

CICS Transaction Server for z/OS
Version 5 Release 6

API-Referenz



Hinweis

Vor Verwendung dieser Informationen und des darin beschriebenen Produkts sollten die Informationen unter [Rechtliche Hinweise zum Produkt](#) gelesen werden.

Diese Ausgabe bezieht sich auf IBM® CICS Transaction Server for z/OS, Version 5 Release 6 (Produktnummer 5655-Y305655-BTA) und alle nachfolgenden Releases und Modifikationen, bis dieser Hinweis in einer Neuausgabe geändert wird.

© Copyright International Business Machines Corporation 1974, 2020.

Inhaltsverzeichnis

Informationen zu diesem PDF-Dokument.....	xi
Kapitel 1. CICS-API-Befehlsformat.....	1
Schreibweise der CICS-Befehlssyntax.....	1
Argumentwerte für CICS-Befehle.....	2
Einschränkungen für CICS-Befehle.....	10
LENGTH-Optionen in CICS-Befehlen.....	10
Option NOHANDLE.....	10
Optionen RESP und RESP2.....	10
Umgesetzter Code für CICS-Befehle.....	11
COBOL-Konvertierungsausgabe.....	11
C-Konvertierungsausgabe.....	12
PL/I-Konvertierungsausgabe.....	12
Assemblerkonvertierungsausgabe.....	13
Kapitel 2. Zusammenfassung der CICS-Befehle.....	17
ABEND.....	27
ACQUIRE.....	28
ADD SUBEVENT.....	31
ADDRESS.....	32
ADDRESS SET.....	34
ALLOCATE (APPC).....	34
ALLOCATE (LUTYPE6.1).....	37
ALLOCATE (MRO).....	39
ASKTIME.....	41
ASSIGN.....	42
Von ASSIGN zurückgegebene Codes.....	58
BIF DEEDIT.....	60
BIF DIGEST.....	61
BUILD ATTACH (LUTYPE6.1).....	62
BUILD ATTACH (MRO).....	65
CANCEL.....	67
CANCEL (BTS).....	69
CHANGE PHRASE.....	71
CHANGE PASSWORD.....	73
CHANGE TASK.....	75
CHECK ACQPROCESS.....	75
CHECK ACTIVITY.....	77
CHECK TIMER.....	80
CONNECT PROCESS.....	81
CONVERSE (Standard).....	84
CONVERSE (APPC).....	84
CONVERSE (LUTYPE2/LUTYPE3).....	84
CONVERSE (LUTYPE4).....	85
CONVERSE (LUTYPE6.1).....	86
CONVERSE (SCS).....	86
CONVERSE (3270 logisch).....	87
CONVERSE (3600-3601).....	87
CONVERSE (3600-3614).....	88
CONVERSE (3650-Interpreter).....	89

CONVERSE (3650-3270).....	89
CONVERSE (3650-3653).....	90
CONVERSE (3650-3680).....	90
CONVERSE (3767).....	91
CONVERSE (3770).....	91
CONVERSE (3790, mit vollem Funktionsumfang oder Anfrage).....	92
CONVERSE (3790- bzw. 3270-Anzeige).....	92
CONVERSE: z/OS Communications Server-Optionen.....	93
CONVERSE (Nicht-z/OS Communications Server-Standard).....	98
CONVERSE (MRO).....	98
CONVERSE (2260).....	99
CONVERSE: Nicht-z/OS Communications Server-Optionen.....	99
CONVERTTIME.....	103
DEFINE ACTIVITY.....	105
DEFINE COMPOSITE EVENT.....	107
DEFINE COUNTER und DEFINE DCOUNTER.....	109
DEFINE INPUT EVENT.....	112
DEFINE PROCESS.....	113
DEFINE TIMER.....	115
DELAY.....	118
DELETE.....	121
DELETE ACTIVITY.....	129
DELETE CHANNEL.....	130
DELETE CONTAINER (BTS).....	131
DELETE CONTAINER (CHANNEL).....	133
DELETE COUNTER und DELETE DCOUNTER.....	134
DELETE EVENT.....	136
DELETE TIMER.....	137
DELETEQ TD.....	137
DELETEQ TS.....	139
DEQ.....	140
DOCUMENT CREATE.....	142
DOCUMENT DELETE.....	146
DOCUMENT INSERT.....	146
DOCUMENT RETRIEVE.....	150
DOCUMENT SET.....	152
DUMP TRANSACTION.....	154
ENDBR.....	159
ENDBROWSE ACTIVITY.....	161
ENDBROWSE CONTAINER (BTS).....	162
ENDBROWSE CONTAINER (CHANNEL).....	162
ENDBROWSE EVENT.....	163
ENDBROWSE PROCESS.....	163
ENDBROWSE TIMER.....	164
ENQ.....	164
ENTER TRACENUM.....	167
EXTRACT ATTACH (LUTYPE6.1).....	169
EXTRACT ATTACH (MRO).....	173
EXTRACT ATTRIBUTES (APPC).....	176
EXTRACT ATTRIBUTES (MRO).....	177
EXTRACT CERTIFICATE.....	178
EXTRACT LOGONMSG.....	181
EXTRACT PROCESS.....	182
EXTRACT TCPIP.....	183
EXTRACT TCT.....	187
EXTRACT WEB.....	188
FETCH ANY.....	193
FETCH CHILD.....	195

FORCE TIMER.....	196
FORMATTIME.....	197
FREE.....	202
FREE (APPC).....	203
FREE (LUTYPE6.1).....	204
FREE (MRO).....	205
FREE CHILD.....	206
FREEMAIN.....	207
FREEMAIN64.....	209
GDS ALLOCATE.....	211
GDS ASSIGN.....	213
GDS CONNECT PROCESS.....	214
GDS EXTRACT ATTRIBUTES.....	217
GDS EXTRACT PROCESS.....	218
GDS FREE.....	219
GDS ISSUE ABEND.....	221
GDS ISSUE CONFIRMATION.....	222
GDS ISSUE ERROR.....	223
GDS ISSUE PREPARE.....	225
GDS ISSUE SIGNAL.....	226
GDS RECEIVE.....	227
GDS SEND.....	229
GDS WAIT.....	232
GET CONTAINER (BTS).....	233
GET CONTAINER (CHANNEL).....	236
GET COUNTER und GET DCOUNTER.....	240
GETMAIN.....	245
GETMAIN64.....	248
GETNEXT ACTIVITY.....	252
GETNEXT CONTAINER (BTS).....	253
GETNEXT CONTAINER (CHANNEL).....	254
GETNEXT EVENT.....	255
GETNEXT PROCESS.....	257
GETNEXT TIMER.....	258
GET64 CONTAINER.....	259
HANDLE ABEND.....	263
HANDLE AID.....	265
HANDLE CONDITION.....	267
IGNORE CONDITION.....	269
INQUIRE ACTIVITYID.....	270
INQUIRE CONTAINER.....	272
INQUIRE EVENT.....	274
INQUIRE PROCESS.....	276
INQUIRE TIMER.....	277
INVOKE APPLICATION.....	278
INVOKE SERVICE.....	282
INVOKE WEBSERVICE.....	287
ISSUE ABEND.....	287
ISSUE ABORT.....	288
ISSUE ADD.....	290
ISSUE CONFIRMATION.....	292
ISSUE COPY (3270 logisch).....	293
ISSUE DISCONNECT (Standard).....	294
ISSUE DISCONNECT (LUTYPE6.1).....	296
ISSUE END.....	296
ISSUE ENDFILE.....	298
ISSUE ENDOUTPUT.....	299
ISSUE EODS.....	299

ISSUE ERASE.....	300
ISSUE ERASEAUP.....	302
ISSUE ERROR.....	303
ISSUE LOAD.....	304
ISSUE NOTE.....	305
ISSUE PASS.....	307
ISSUE PREPARE.....	308
ISSUE PRINT.....	310
ISSUE QUERY.....	311
ISSUE RECEIVE.....	312
ISSUE REPLACE.....	313
ISSUE RESET.....	316
ISSUE SEND.....	316
ISSUE SIGNAL (APPC).....	318
ISSUE SIGNAL (LUTYPE6.1).....	319
ISSUE WAIT.....	320
JOURNAL.....	322
LINK.....	322
LINK ACQPROCESS.....	331
LINK ACTIVITY.....	334
LOAD.....	337
MONITOR.....	340
MOVE CONTAINER (BTS).....	342
MOVE CONTAINER (CHANNEL).....	345
POINT.....	347
POP HANDLE.....	348
POST.....	348
PURGE MESSAGE.....	352
PUSH HANDLE.....	353
PUT CONTAINER (BTS).....	353
PUT CONTAINER (CHANNEL).....	355
PUT64 CONTAINER.....	359
QUERY CHANNEL.....	363
QUERY COUNTER und QUERY DCOUNTER.....	364
QUERY SECURITY.....	367
READ.....	374
Befehl READNEXT.....	387
READPREV.....	398
READQ TD.....	408
READQ TS.....	411
RECEIVE (Standardformat für z/OS Communications Server).....	414
RECEIVE (APPC).....	415
RECEIVE (LUTYPE2/LUTYPE3).....	415
RECEIVE (LUTYPE4).....	416
RECEIVE (LUTYPE6.1).....	416
RECEIVE (logische 3270-Einheit).....	417
RECEIVE (3600-Pipeline).....	417
RECEIVE (3600-3601).....	418
RECEIVE (3600-3614).....	418
RECEIVE (3650).....	419
RECEIVE (logische 3767-Einheit).....	420
RECEIVE (3770).....	420
RECEIVE (3790 mit vollem Funktionsumfang oder Anfrage).....	421
RECEIVE: Optionen für z/OS Communications Server.....	421
RECEIVE (Standardformat ohne z/OS Communications Server).....	424
RECEIVE (MRO).....	425
RECEIVE (2260).....	425
RECEIVE (2980).....	426

RECEIVE (3790- bzw. 3270-Anzeige).....	428
RECEIVE: Optionen ohne z/OS Communications Server	429
RECEIVE MAP.....	432
RECEIVE MAP MAPPINGDEV.....	435
RECEIVE PARTN.....	437
RELEASE.....	439
REMOVE SUBEVENT.....	441
REQUEST ENCRYPTPTKT.....	442
REQUEST PASSTICKET.....	444
RESET ACQPROCESS.....	446
RESET ACTIVITY.....	447
RESETBR.....	449
RESUME.....	453
RETRIEVE.....	455
RETRIEVE REATTACH EVENT.....	458
RETRIEVE SUBEVENT.....	459
RETURN.....	460
REWIND COUNTER und REWIND DCOUNTER.....	464
REWRITE.....	467
ROUTE.....	472
RUN.....	476
RUN TRANSID.....	481
SEND (Standardformat für z/OS Communications Server).....	482
SEND (APPC).....	483
SEND (LUTYPE2/LUTYPE3).....	483
SEND (LUTYPE4).....	484
SEND (LUTYPE6.1).....	484
SEND (SCS).....	485
SEND (3270 logisch).....	485
SEND (3600-Pipeline).....	486
SEND (3600-3601).....	486
SEND (3600-3614).....	487
SEND (3650-Interpreter).....	488
SEND (3650-3270).....	488
SEND (3650-3653).....	489
SEND (logische 3650- bzw. 3680-Einheit).....	489
SEND (3767).....	490
SEND (3770).....	490
SEND (3790 mit vollem Funktionsumfang oder Anfrage).....	491
SEND (3790 SCS).....	491
SEND (3790- bzw. 3270-Anzeige).....	491
SEND (3790- bzw. 3270-Drucker).....	492
SEND: Optionen für z/OS Communications Server.....	492
SEND (Standardformat ohne z/OS Communications Server).....	496
SEND (MRO).....	496
SEND (2260).....	497
SEND (2980).....	497
SEND: Optionen für Nicht-z/OS Communications Server.....	497
SEND CONTROL.....	500
SEND MAP.....	505
SEND MAP MAPPINGDEV.....	513
SEND PAGE.....	516
SEND PARTNSET.....	519
SEND TEXT.....	519
SEND TEXT MAPPED.....	526
SEND TEXT NOEDIT.....	528
SIGNAL EVENT.....	531
SIGNOFF.....	532

SIGNON.....	533
SIGNON TOKEN.....	538
SOAPFAULT ADD.....	543
SOAPFAULT CREATE.....	545
SOAPFAULT DELETE.....	549
SPOOLCLOSE.....	550
SPOOLOPEN INPUT.....	552
SPOOLOPEN OUTPUT.....	555
SPOOLREAD.....	560
SPOOLWRITE.....	562
START.....	565
START ATTACH.....	575
START BREXIT.....	576
START CHANNEL.....	579
STARTBR.....	583
STARTBROWSE ACTIVITY.....	590
STARTBROWSE CONTAINER (BTS).....	591
STARTBROWSE CONTAINER (CHANNEL).....	593
STARTBROWSE EVENT.....	594
STARTBROWSE PROCESS.....	595
STARTBROWSE TIMER.....	596
SUSPEND.....	597
SUSPEND (BTS).....	597
SYNCPOINT.....	599
SYNCPOINT ROLLBACK.....	600
TEST EVENT.....	601
TRANSFORM DATATOJSON.....	602
TRANSFORM DATATOXML.....	603
TRANSFORM JSONTODATA.....	606
TRANSFORM XMLTODATA.....	608
UNLOCK.....	611
UPDATE COUNTER und UPDATE DCOUNTER.....	615
VERIFY PASSWORD.....	618
VERIFY PHRASE.....	621
VERIFY TOKEN.....	625
WAIT CONVID (APPC).....	629
WAIT EVENT.....	630
WAIT EXTERNAL.....	632
WAIT JOURNALNAME.....	635
WAIT JOURNALNUM.....	637
WAIT SIGNAL.....	637
WAIT TERMINAL.....	638
WAITCICS.....	639
WEB CLOSE.....	642
WEB CONVERSE.....	644
WEB ENDBROWSE FORMFIELD.....	658
WEB ENDBROWSE HTTPHEADER.....	659
WEB ENDBROWSE QUERYPARM.....	660
WEB EXTRACT.....	660
WEB OPEN.....	666
WEB PARSE URL.....	671
WEB READ FORMFIELD.....	674
WEB READ HTTPHEADER.....	676
WEB READ QUERYPARM.....	678
WEB READNEXT FORMFIELD.....	680
WEB READNEXT HTTPHEADER.....	681
WEB READNEXT QUERYPARM.....	683
WEB RECEIVE (Server).....	684

WEB RECEIVE (Client).....	691
WEB RETRIEVE.....	698
WEB SEND (Server).....	699
WEB SEND (Client).....	708
WEB STARTBROWSE FORMFIELD.....	720
WEB STARTBROWSE HTTPHEADER.....	721
WEB STARTBROWSE QUERYPARM.....	722
WEB WRITE HTTPHEADER.....	724
WRITE.....	727
WRITE JOURNALNAME.....	734
WRITE JOURNALNUM.....	738
WRITE OPERATOR.....	738
WRITEQ TD.....	741
WRITEQ TS.....	743
WSACONTEXT BUILD.....	747
WSACONTEXT DELETE.....	752
WSACONTEXT GET.....	752
WSAEPR CREATE.....	756
XCTL.....	759
Kapitel 3. Threadsichere Befehle.....	765
Kapitel 4. Von allen Befehlen verwendete Datenbereiche für CICS-Werte.....	769
CVDAs und numerische Werte in alphabetischer Reihenfolge.....	769
CVDA-Werte für die Option DEVICE.....	798
CVDAs und numerische Werte in numerischer Reihenfolge.....	801
Kapitel 5. Landessprachencodes für die Anwendungsentwicklung.....	831
Kapitel 6. Terminalsteuerung.....	833
Befehle und Optionen für Terminals und logische Einheiten.....	833
Programmierung für Fernschreiber.....	835
Operationen für Bildschirmseinheiten.....	835
Angezeigte Informationen drucken (ISSUE PRINT).....	836
Angezeigte Informationen kopieren (ISSUE COPY).....	836
Alle ungeschützten Felder entfernen (ISSUE ERASEAUP).....	837
Eingabe ohne Daten verarbeiten (RECEIVE).....	837
Kapitel 7. Common Programming Interface Communications (CPI Communicati- ons).....	839
SAA Resource Recovery.....	839
Kapitel 8. BMS-spezifische Konstanten.....	841
Steuerwertkonstanten für Magnetcodeleser (DFHMSRCA).....	845
Werte für MSR-Steuerbyte.....	845
AID-Zeichenkonstanten, DFHAID.....	846
Kapitel 9. BMS-Makros.....	849
Maskengruppen-, Masken- und Felddefinition.....	849
Definition von Partitionsgruppen.....	850
Feldgruppen.....	850
DFHMDF.....	851
DFHMDI.....	862
DFHMSD.....	871
DFHPDI.....	881
DFHPSD.....	882

Bemerkungen.....	885
Index.....	891

Informationen zu diesem PDF-Dokument

Dieses PDF-Dokument enthält Referenzinformationen zu den Befehlen der CICS-Anwendungsprogrammierschnittstelle (API). Diese Dokumentation wendet sich an Anwendungsprogrammierer, die Anwendungen für die Interaktion mit CICS schreiben. Vor CICS TS Version 5.4 trug dieses PDF-Dokument die Bezeichnung *Application Programming Reference*.

Informationen dazu, wie Anwendungen für CICS mithilfe dieser API geschrieben werden, finden Sie in der Veröffentlichung *Developing CICS Applications*.

Details zu den Begriffen und zur Notation, die in diesem Handbuch verwendet werden, finden Sie unter [In der Dokumentation zu CICS verwendete Konventionen und Terminologie](#) im IBM Knowledge Center.

Datum dieses PDF-Dokuments

Dieses PDF-Dokument wurde am 28. Mai 2020 erstellt.

Kapitel 1. CICS-API-Befehlsformat

Das allgemeine Format eines CICS-Befehls besteht aus der Angabe EXECUTE CICS (oder EXEC CICS), auf die der Name des gewünschten **Befehls** sowie gegebenenfalls eine oder mehrere **Optionen** folgen.

Das Befehlsformat sieht wie folgt aus:

```
EXEC CICS befehl option(arg)....
```

Dabei gilt:

befehl

Beschreibt die erforderliche Operation (Beispiel: READ).

option

Beschreibt eine der zahlreichen optionalen Einrichtungen, die für jede Funktion verfügbar sind. Einige Optionen haben ein nachfolgendes Argument in Klammern. Sie können Optionen (einschließlich Optionen mit erforderlichen Argumenten) in beliebiger Reihenfolge angeben.

arg

(kurz für 'argument') ist ein Wert wie "datenwert" oder "name". Eine Angabe "datenwert" kann eine Konstante sein, das heißt, dass ein Argument, das Daten an CICS sendet, im Allgemeinen ein "Datenwert" ist. Ein Argument, das Daten von CICS empfängt, muss ein "Datenbereich" sein.

Einige Argumente, die als "Datenbereich" beschrieben werden, können Daten sowohl senden als auch empfangen. In diesen Fällen müssen Sie sicherstellen, dass sich der "Datenbereich" nicht im geschützten Speicher befindet.

Ein Beispiel für einen CICS-Befehl sieht wie folgt aus:

```
EXEC CICS READ
      FILE('FILEA')
      INTO(FILEA)
      RIDFLD(KEYNUM)
      UPDATE
```

Sie müssen den entsprechenden Befehlsabschluss hinzufügen (siehe „Schreibweise der CICS-Befehlssyntax“ auf Seite 1).

Anmerkung: Wenn Sie Kommentare für CICS-Befehle hinzufügen wollen, können Sie dies, jedoch nur in Assembler, indem Sie einen Punkt oder ein Komma als Begrenzungszeichen nach dem letzten Argument verwenden. Beispiel:

```
EXEC CICS ADDRESS EIB(MYUEIB),      @F1A
```

Wenn ein Punkt oder ein Komma mit einem EXEC CICS-Befehl verwendet wird, muss die folgende Zeile zwischen Spalte 2 und Spalte 16 beginnen und die Spalte 72 ein Fortsetzungszeichen enthalten. Die folgende Zeile darf nicht nach Spalte 17 beginnen. Wenn kein Komma oder Punkt hinzugefügt wird, muss die folgende Zeile in oder nach Spalte 2 beginnen und in Spalte 71 enden, wobei das Fortsetzungszeichen in Spalte 72 angegeben wird.

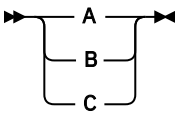
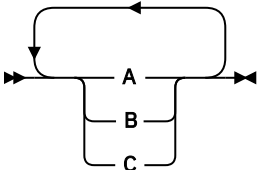
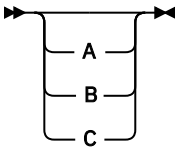
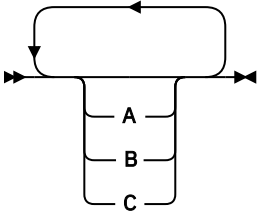
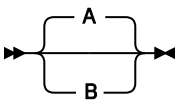
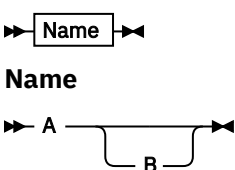
Schreibweise der CICS-Befehlssyntax

In der CICS-Dokumentation werden CICS-Befehle auf standardisierte Weise dargestellt. Zur Interpretation der Syntax folgen Sie den Pfeilen von links nach rechts.

Die Zeichenfolge 'EXEC CICS', die stets dem Schlüsselwort jedes Befehls vorausgeht, ist in der Syntax ebensowenig enthalten wie die in COBOL verwendete Anweisung 'END-EXEC' oder das in PL/I und C ver-

wendete Semikolon (;), das Sie am Ende jedes CICS-Befehls codieren müssen. In der Programmiersprache C können Nullzeichen als Markierung für das Zeichenfolgende verwendet werden, was von CICS jedoch nicht erkannt wird. Daher darf mitten in einer Codierungszeile auf ein Komma oder einen Punkt nie ein Leerzeichen (X'40') folgen.

Es gelten die folgenden Konventionen:

Symbol	Aktion
	Eine Gruppe von Alternativen, von denen Sie eine codieren müssen .
	Eine Gruppe von Alternativen, von denen Sie eine codieren müssen . Sie können mehrere dieser Alternativen in einer beliebigen Reihenfolge codieren.
	Eine Gruppe von Alternativen, von denen Sie eine codieren können .
	Eine Gruppe von Alternativen, von denen Sie beliebig viele (auch gar keine) in einer beliebigen Reihenfolge codieren können.
	Zwei Alternativen, von denen A der Standardwert ist.
	Wird für den genannten Abschnitt anstelle des Namens verwendet.
Interpunktion und Großbuchstaben	Nehmen Sie die Codierung exakt wie dargestellt vor.
Kleinbuchstaben	Codieren Sie Ihren eigenen Text wie erforderlich (z. B. name).

Argumentwerte für CICS-Befehle

Die Daten, die einer Befehlsoption zugeordnet werden, werden als *Argument* der Option bezeichnet. Jeder Typ von Argument kann verschiedene Datentypen enthalten. Einige Argumente geben Informationen aus CICS an das Programm zurück, während andere durch das Programm festgelegt werden.

Optionen in einem CICS-Befehl können die folgenden Argumentwerte haben:

- *Datenwert*
- *Datenbereich*
- *CVDA* (Datenbereich für CICS-Werte)
- *Zeigerwert*
- *Zeigerverweis*
- *Name*
- *Dateiname*
- *Systemname*
- *Bezeichnung (Label)*
- *hhmmss*

Nur für AMODE(64)-Programme können Optionen in einem CICS-Befehl außerdem die folgenden Argumentwerte haben:

- *Datenbereich64*
- *Zeigerwert64*
- *Zeigerverweis64*

Datenbereiche und Datenwerte

Datenbereiche und Datenwerte sind die grundlegenden Argumenttypen. Der Unterschied zwischen ihnen ist die Richtung, in der Informationen fließen, wenn eine Task einen Befehl ausführt. Ein Argument *datenwert* ist immer ein Sender und kann nur ein Sender sein. Es übermittelt Daten an CICS, die von CICS zur Verarbeitung des Befehls verwendet werden. Ein Argument *datenbereich* ist ein Empfänger, der von CICS zur Rückgabe von Informationen an den Aufrufer verwendet wird. Ein Argument *datenbereich* kann außerdem ein Sender sein, zum Beispiel wenn die Daten, die an CICS zu übermitteln sind, eine variable Länge haben (wie in der Option FROM) oder wenn ein Feld sowohl für die Eingabe als auch für die Ausgabe verwendet wird.

COBOL-Argumentwerte

Die Argumentwerte können wie folgt ersetzt werden:

- Ein Argument *datenwert* kann durch einen beliebigen COBOL-Datennamen des korrekten Datentyps oder durch eine Konstante ersetzt werden, die in den korrekten Typ für das Argument konvertiert werden kann. In der folgenden Tabelle wird gezeigt, wie der korrekte Datentyp definiert wird:

Datentyp	COBOL-Definition
Halbwort-Binärwert	PIC S9(4) COMP
Vollwort-Binärwert	PIC S9(8) COMP
Doppelwort-Binärwert <i>ohne Vorzeichen</i>	PIC 9(18) COMP
Zeichenfolge	PIC X(n) mit n = Anzahl Byte
UTF-8-Zeichenfolge	PIC X(n) mit n = Anzahl Byte

- Ein Argument *datenbereich* kann durch einen beliebigen COBOL-Datennamen des korrekten Datentyps für das Argument ersetzt werden. In der folgenden Tabelle wird gezeigt, wie der korrekte Datentyp definiert wird:

Datentyp	COBOL-Definition
Halbwort-Binärwert	PIC S9(4) COMP
Vollwort-Binärwert	PIC S9(8) COMP

Datentyp	COBOL-Definition
Doppelwort-Binärwert <i>ohne Vorzeichen</i>	PIC 9(18) COMP
Zeichenfolge	PIC X(n) mit n = Anzahl Byte
UTF-8-Zeichenfolge	PIC X(n) mit n = Anzahl Byte

Wenn der Datentyp nicht angegeben ist, kann sich *datenbereich* auf ein elementares Element oder ein Gruppenelement beziehen.

- Werte für *cvda* werden in Datenbereiche für CICS-Werte (CVDAs) beschrieben.
- Ein Argument *zeigerwert* kann durch eine Zeigervariable oder durch das Sonderregister ADDRESS ersetzt werden.
- Ein Argument *zeigerverweis* kann durch eine Zeigervariable oder durch das Sonderregister ADDRESS ersetzt werden.
- Ein Argument *name* kann durch jeden der folgenden Werte ersetzt werden:
 - Eine Zeichenfolge, die als alphanumerisches Literal angegeben wird. Wenn diese Zeichenfolge kürzer als die erforderliche Länge ist, wird sie mit Leerzeichen aufgefüllt.
 - Ein COBOL-Datenbereich mit einer Länge, die gleich der erforderlichen Länge für den Namen ist. Der Wert in *datenbereich* ist der Name, der von dem Argument verwendet werden soll. Wenn *datenbereich* kürzer als die erforderliche Länge ist, sind die Überschusszeichen nicht definiert, was zu unvorhersehbaren Ergebnissen führen kann.

Ein Argument *dateiname*, wie es in FILE(*dateiname*) verwendet wird, gibt den Namen der Datei an. Der Name muss 1 - 8 Zeichen aus dem Bereich A–Z, 0–9, \$, @ und # enthalten.

Ein Argument *systemname*, wie es in SYSID(*systemname*) verwendet wird, gibt den Namen des Systems an, an das die Anforderung gerichtet wird. Der Name muss 1–4 Zeichen aus dem Bereich A–Z, 0–9, \$, @ und # enthalten.

- Ein Argument *bezeichnung* kann durch einen beliebigen COBOL-Absatznamen oder Abschnittsnamen ersetzt werden.
- Ein Argument *hhmmss* kann durch eine Dezimalkonstante oder einen Datennamen der Form PIC S9(7) COMP-3 ersetzt werden. Der muss die Form 0HHMMSS+ haben, wobei Folgendes gilt:

HH

Stellt Stunden von 00 bis 99 dar.

MM

Stellt Minuten von 00 bis 59 dar.

SS

Stellt Sekunden von 00 bis 59 dar.

In COBOL müssen Sie die Option LENGTH nur codieren, wenn das Programm Daten einer Länge lesen oder schreiben soll, die von der referenzierten Variablen abweicht.

C-Argumentwerte

Die Argumentwerte können wie folgt ersetzt werden:

- Ein Argument *datenwert* kann durch einen beliebigen Ausdruck in der Programmiersprache C ersetzt werden, der in den korrekten Datentyp für das Argument konvertiert werden kann. In der folgenden Tabelle wird gezeigt, wie der korrekte Datentyp definiert wird:

Datentyp	C-Definition
Halbwort-Binärwert	short int
Vollwort-Binärwert	long int
Doppelwort-Binärwert	char[8]

Datentyp	C-Definition
Zeichenfolge	char[n] mit n = Anzahl Byte
UTF-8-Zeichenfolge	char[n] mit n = Anzahl Byte

Ein Argument *datenwert* schließt einen Wert *datenbereich* als Teilmenge ein.

- Ein Argument *datenbereich* kann durch eine beliebige C-Datenreferenz ersetzt werden, die den korrekten Datentyp für das Argument hat. In der folgenden Tabelle wird gezeigt, wie der korrekte Datentyp definiert wird:

Datentyp	C-Definition
Halbwort-Binärwert	short int
Vollwort-Binärwert	long int
Doppelwort-Binärwert	char[8]
Zeichenfolge	char[n] mit n = Anzahl Byte
UTF-8-Zeichenfolge	char[n] mit n = Anzahl Byte

Wenn der Datentyp nicht angegeben wird, kann sich *datenbereich* auf einen skalaren Datentyp, auf ein Array oder eine Struktur beziehen. Der Verweis muss auf zusammenhängenden Speicher verweisen.

- Werte für *cvda* werden in Datenbereiche für CICS-Werte (CVDAs) beschrieben.
- Ein Argument *zeigerwert* (das *zeigerverweis* als Teilmenge enthält) kann durch einen beliebigen Ausdruck in der Programmiersprache C ersetzt werden, der in eine Adresse konvertiert werden kann.
- Ein Argument *zeigerverweis* kann durch einen beliebigen C-Zeigertypverweis ersetzt werden.
- Ein Argument *name* kann durch jeden der folgenden Werte ersetzt werden:
 - Eine Zeichenfolge in doppelten Anführungszeichen (d. h. eine Literalkonstante).
 - Ein C-Ausdruck oder -Verweis, dessen Wert in ein Zeichenarray mit einer Länge konvertiert werden kann, die gleich der maximal zulässigen Länge für den Namen ist. Der Wert des Zeichenarrays ist der Name, der von dem Argument verwendet werden soll.

Ein Argument *dateiname*, wie es in FILE(*dateiname*) verwendet wird, gibt den Namen der Datei an. Der Name muss 1 - 8 Zeichen aus dem Bereich A-Z, 0-9, \$, @ und # enthalten.

Ein Argument *systemname*, wie es in SYSID(*systemname*) verwendet wird, gibt den Namen des Systems an, an das die Anforderung gerichtet wird. Der Name muss 1 - 4 Zeichen aus dem Bereich A-Z, 0-9, \$, @ und # enthalten.

- Ein Argument *bezeichnung* wird in der Programmiersprache C nicht unterstützt.
- Ein Argument *hhmmss* kann durch eine ganzzahlige Konstante ersetzt werden. Andernfalls muss die Anwendung sicherstellen, dass der Wert, der an CICS übergeben wird, ein gepacktes Dezimalformat hat. Die Sprache stellt keinen gepackten Dezimaltyp bereit.

HH

Stellt Stunden von 00 bis 99 dar.

MM

Stellt Minuten von 00 bis 59 dar.

SS

Stellt Sekunden von 00 bis 59 dar.

Viele Befehle sind mit der Übertragung von Daten zwischen dem Anwendungsprogramm und CICS verbunden. In den meisten Fällen muss bei Verwendung von SET die Option LENGTH verwendet werden. Die Syntax der einzelnen Befehle und der zugehörigen Optionen zeigt, ob diese Regel gilt.

PL/I-Argumentwerte

Die Argumentwerte können wie folgt ersetzt werden:

- Ein Argument *datenwert* kann durch einen beliebigen PL/I-Ausdruck ersetzt werden, der in den korrekten Datentyp für das Argument konvertiert werden kann. In der folgenden Tabelle wird gezeigt, wie der korrekte Datentyp definiert wird:

Datentyp	PL/I-Definition
Halbwort-Binärwert	FIXED BIN(15)
Vollwort-Binärwert	FIXED BIN(31)
Doppelwort-Binärwert	CHAR (8)
Zeichenfolge	CHAR(<i>n</i>) mit <i>n</i> = Anzahl Byte
UTF-8-Zeichenfolge	CHAR(<i>n</i>) mit <i>n</i> = Anzahl Byte

Ein Argument *datenwert* schließt einen Wert *datenbereich* als Teilmenge ein.

- Ein Argument *datenbereich* kann durch eine beliebige PL/I-Datenreferenz ersetzt werden, die den korrekten Datentyp für das Argument hat. In der folgenden Tabelle wird gezeigt, wie der korrekte Datentyp definiert wird:

Datentyp	PL/I-Definition
Halbwort-Binärwert	FIXED BIN(15)
Vollwort-Binärwert	FIXED BIN(31)
Doppelwort-Binärwert	CHAR (8)
Zeichenfolge	CHAR(<i>n</i>) mit <i>n</i> = Anzahl Byte
UTF-8-Zeichenfolge	CHAR(<i>n</i>) mit <i>n</i> = Anzahl Byte

Wenn der Datentyp nicht angegeben wird, kann sich *datenbereich* auf ein Element, ein Array oder eine Struktur beziehen. Beispiel: FROM(P->STRUCTURE) LENGTH(LNG). Der Verweis muss auf verbundenen Speicher verweisen.

Der Datenbereich muss außerdem das korrekte PL/I-Ausrichtungsattribut haben: ALIGNED für binäre Elemente und UNALIGNED für Zeichenfolgen.

Wenn Sie eine variierende Datenzeichenfolge mit expliziter Länge verwenden, beginnen die übergebenen Daten mit einem zwei Byte langen Feld und ihre Länge ist die maximale Länge, die für die Zeichenfolge deklariert ist. Wenn Sie eine Länge im Befehl angeben, haben die übergebenen Daten diese Länge. Das heißt, das zwei Byte lange Feld, gefolgt von Daten bis zur angegebenen Länge.

- Werte für *cvda* werden in [Datenbereiche für CICS-Werte \(CVDAs\)](#) beschrieben.
- Ein Argument *zeigerwert* (das *zeigerverweis* als Teilmenge enthält) kann durch einen beliebigen PL/I-Ausdruck ersetzt werden, der in einen Zeiger (POINTER) konvertiert werden kann.
- Ein Argument *zeigerverweis* kann durch einen beliebigen PL/I-Verweis vom Typ POINTER ALIGNED ersetzt werden.
- Ein Argument *name* kann durch jeden der folgenden Werte ersetzt werden:
 - Eine Zeichenfolge in einfachen Anführungszeichen (d. h. eine Literalkonstante).
 - Ein PL/I-Ausdruck oder -Verweis, dessen Wert in eine Zeichenfolge mit einer Länge konvertiert werden kann, die gleich der maximal zulässigen Länge für den Namen ist. Der Wert der Zeichenfolge ist der Name, der von dem Argument verwendet werden soll.

Ein Argument *dateiname*, wie es in FILE(*dateiname*) verwendet wird, gibt den Namen der Datei an. Der Name muss 1-8 Zeichen aus dem Bereich A-Z, 0-9, \$, @ und # enthalten.

Ein Argument *systemname*, wie es in `SYSID(systemname)` verwendet wird, gibt den Namen des Systems an, an das die Anforderung gerichtet wird. Der Name muss 1-4 Zeichen aus dem Bereich A–Z, 0–9, \$, @ und # enthalten.

- Ein Argument *bezeichnung* kann durch einen beliebigen PL/I-Ausdruck ersetzt werden, dessen Wert eine Bezeichnung (Label) ist.
- Ein Argument *hhmmss* kann durch eine Dezimalkonstante oder einen Ausdruck ersetzt werden, der in den Typ `FIXED DECIMAL(7,0)` konvertiert werden kann. Der muss die Form `0HHMMSS+` haben, wobei Folgendes gilt:

HH

Stellt Stunden von 00 bis 99 dar.

MM

Stellt Minuten von 00 bis 59 dar.

SS

Stellt Sekunden von 00 bis 59 dar.

Wenn das Attribut `UNALIGNED` in den `ENTRY`-Deklarationen hinzugefügt wird, die durch den CICS-Umsetzer mit einer Anweisung `DEFAULT DESCRIPTORS` generiert werden, müssen die Argumente 'datenbereich' und 'zeigerverweis' für CICS-Befehle ebenfalls `UNALIGNED` sein. Analog müssen für das Attribut `ALIGNED` die Argumente 'datenbereich' oder 'zeigerverweis' ebenfalls `ALIGNED` sein.

Viele Befehle sind mit der Übertragung von Daten zwischen dem Anwendungsprogramm und CICS verbunden. In den meisten Fällen muss die Länge der Daten, die übertragen werden sollen, durch das Anwendungsprogramm bereitgestellt werden. Wenn ein Datenbereich jedoch als Quelle oder Ziel angegeben wird, ist es nicht erforderlich, die Länge explizit anzugeben, da der Befehlssprachenumsetzer einen geeigneten Standardlängenwert, d. h. `STG(datenbereich)` oder `CSTG(datenbereich)`, generiert.

Argumentwerte der Assemblersprache für **AMODE(24)-** und **AMODE(31)-Programme**

Im Allgemeinen kann ein Argument entweder die Adresse der Daten oder die Daten selbst sein (in der Terminologie der Assemblersprache: entweder ein verschiebbarer Ausdruck oder ein absoluter Ausdruck).

Ein verschiebbarer Ausdruck ('relocatable expression') darf keine unpaarigen eckigen Klammern (außerhalb von Anführungszeichen) oder unpaarige Anführungszeichen (außer bei Verweisen auf Längenattribute) enthalten. Wenn diese Regel eingehalten wird, kann ein beliebiger Ausdruck verwendet werden, einschließlich Literalkonstanten wie `=AL2(100)`, Formate wie `20(0,R11)` sowie Formate, die die Einrichtungen zum Ersetzen von Makros verwenden.

Ein absoluter Ausdruck muss ein einzelner Term sein, der entweder ein Längenattributverweis oder eine selbstdefinierende Konstante ist.

Verwenden Sie (mit `EQU`) gleichgesetzte Symbole mit Sorgfalt. Solche Symbole sollten nur verwendet werden, wenn auf Register verwiesen wird (Zeigerverweise). Wenn ein gleichgesetztes Symbol zum Beispiel für eine Länge verwendet wird, wird es als Adresse der Länge behandelt und es tritt ein unvorhersehbarer Fehler auf.

Für `AMODE(24)-` und `AMODE(31)-`Assemblerprogramme können Argumentwerte wie folgt ersetzt werden:

- Ein Argument *datenwert* kann durch einen beliebigen verschiebbaren Ausdruck, bei dem es sich in der Assemblersprache einen Verweis auf Daten des korrekten Datentyps für das Argument oder durch eine Konstante des korrekten Typs für das Argument ersetzt werden.
- Ein Argument *datenbereich* kann durch einen verschiebbaren Ausdruck ersetzt werden, bei dem es sich in der Assemblersprache um einen Verweis auf Daten des korrekten Datentyps für das Argument handelt.
- Werte für *cvda* werden in Datenbereiche für CICS-Werte (CVDAs) beschrieben.
- Ein Argument *zeigerwert* kann durch einen absoluten Ausdruck ersetzt werden, bei dem es sich in der Assemblersprache um einen Verweis auf ein Register handelt.

- Ein Argument *zeigerverweis* kann durch einen absoluten Ausdruck ersetzt werden, bei dem es sich in der Assemblersprache um einen Verweis auf ein Register handelt.
- Ein Argument *name* kann entweder durch eine Zeichenfolge in einfachen Anführungszeichen oder durch einen verschiebbaren Ausdruck, der auf eine Zeichenfolge verweist, ersetzt werden. Die Länge ist gleich der maximal zulässigen Länge für den Namen. Der Wert der Zeichenfolge ist der Name, der von dem Argument verwendet werden soll.

Ein Argument *dateiname*, wie es in FILE(*dateiname*) verwendet wird, gibt den Namen der Datei an. Der Name muss 1 - 8 Zeichen aus dem Bereich A–Z, 0–9, \$, @ und # enthalten.

Ein Argument *systemname*, wie es in SYSID(*systemname*) verwendet wird, gibt den Namen des Systems an, an das die Anforderung gerichtet wird. Der Name muss 1 - 4 Zeichen aus dem Bereich A–Z, 0–9, \$, @ und # enthalten.

- Ein Argument *bezeichnung* (Label) gibt eine Zieladresse an, an die die Steuerung übertragen wird. Es kann durch die Bezeichnung der Zielanweisung oder durch die Bezeichnung einer Adresskonstanten für das Ziel ersetzt werden. Diese Konstante darf keine Länge angeben.

Sie können auch den Ausdruck =A(*ziel*) verwenden, wobei *ziel* ein verschiebbarer Ausdruck ist, der das Ziel bezeichnet.

Die folgenden Befehle sind zum Beispiel äquivalent:

```
HANDLE CONDITION ERROR(DEST)
HANDLE CONDITION ERROR(ADCON)
HANDLE CONDITION ERROR(=A(DEST))
:
DEST BR 14
ADCON DC A(DEST)
```

- Die Angabe *hhmmss* kann durch eine selbstdefinierende Dezimalkonstante oder durch einen Verweis in Assemblersprache auf ein Feld, das als PL4 definiert ist, ersetzt werden. Der muss die Form 0HHMMSS + haben, wobei Folgendes gilt:

HH

Stellt Stunden von 00 bis 99 dar.

MM

Stellt Minuten von 00 bis 59 dar.

SS

Stellt Sekunden von 00 bis 59 dar.

Viele Befehle sind mit der Übertragung von Daten zwischen dem Anwendungsprogramm und CICS verbunden. In den meisten Fällen muss das Anwendungsprogramm die Länge der Daten, die übertragen werden sollen, angeben. Wenn ein Datenbereich jedoch als Quelle oder Ziel angegeben wird, ist es nicht erforderlich, die Länge explizit anzugeben, da der Befehlssprachenumsetzer eine Standardlänge generiert. Beispiel:

```
xxx DC CL8
.
EXEC CICS ... LENGTH(L'xxx)
```

Argumentwerte der Assemblersprache für AMODE(64)-Programme

Im Allgemeinen kann ein Argument entweder die Adresse der Daten oder die Daten selbst sein (in der Terminologie der Assemblersprache: entweder ein verschiebbarer Ausdruck oder ein absoluter Ausdruck).

Ein verschiebbarer Ausdruck ('relocatable expression') darf keine unpaarigen eckigen Klammern (außerhalb von Anführungszeichen) oder unpaarige Anführungszeichen (außer bei Verweisen auf Längenattribute) enthalten. Wenn diese Regel eingehalten wird, kann ein beliebiger Ausdruck verwendet werden, einschließlich Literalkonstanten wie =AL2(100), Formate wie 20(0,R11) sowie Formate, die die Einrichtungen zum Ersetzen von Makros verwenden.

Ein absoluter Ausdruck muss ein einzelner Term sein, der entweder ein Längenattributverweis oder eine selbstdefinierende Konstante ist.

Verwenden Sie (mit EQU) gleichgesetzte Symbole mit Sorgfalt. Solche Symbole sollten nur verwendet werden, wenn auf Register verwiesen wird (Zeigerverweise). Wenn ein gleichgesetztes Symbol zum Beispiel für eine Länge verwendet wird, wird es als Adresse der Länge behandelt und es tritt ein unvorhersehbarer Fehler auf.

Für AMODE(64)-Assemblerprogramme, über Language Environment (LE) ausgeführt werden, können Argumentwerte wie folgt ersetzt werden:

- Ein Argument *datenwert* kann durch einen beliebigen verschiebbaren Ausdruck, bei dem es sich in der Assemblersprache einen Verweis auf Daten des korrekten Datentyps für das Argument oder durch eine Konstante des korrekten Typs für das Argument ersetzt werden.
- Ein Argument *datenbereich* kann durch einen verschiebbaren Ausdruck ersetzt werden, bei dem es sich in der Assemblersprache um einen Verweis auf Daten des korrekten Datentyps für das Argument handelt.
- Ein Argument *datenbereich64* durch einen verschiebbaren Ausdruck, bei dem es sich in der Assemblersprache um einen 64-Bit-Verweis auf Daten des korrekten Datentyps für das Argument handelt.
- Werte für *cvda* werden in Datenbereiche für CICS-Werte (CVDAs) beschrieben.
- Ein Argument *zeigerwert* kann durch einen absoluten Ausdruck ersetzt werden, bei dem es sich in der Assemblersprache um einen Verweis auf ein Register handelt.
- Ein Argument *zeigerwert64* kann durch einen absoluten Ausdruck ersetzt werden, bei dem es sich in der Assemblersprache um einen 64-Bit-Verweis auf ein Register handelt.
- Ein Argument *zeigerverweis* kann durch einen absoluten Ausdruck ersetzt werden, bei dem es sich in der Assemblersprache um einen Verweis auf ein Register handelt.
- Ein Argument *zeigerverweis64* kann durch einen absoluten Ausdruck ersetzt werden, bei dem es sich in der Assemblersprache um einen 64-Bit-Verweis auf ein Register handelt.
- Ein Argument *name* kann entweder durch eine Zeichenfolge in einfachen Anführungszeichen oder durch einen verschiebbaren Ausdruck, der auf eine Zeichenfolge verweist, ersetzt werden. Die Länge ist gleich der maximal zulässigen Länge für den Namen. Der Wert der Zeichenfolge ist der Name, der von dem Argument verwendet werden soll.

Ein Argument *dateiname*, wie es in FILE(*dateiname*) verwendet wird, gibt den Namen der Datei an. Der Name muss 1 - 8 Zeichen aus dem Bereich A-Z, 0-9, \$, @ und # enthalten.

Ein Argument *systemname*, wie es in SYSID(*systemname*) verwendet wird, gibt den Namen des Systems an, an das die Anforderung gerichtet wird. Der Name muss 1 - 4 Zeichen aus dem Bereich A-Z, 0-9, \$, @ und # enthalten.

- Die Angabe *hhmmss* kann durch eine selbstdefinierende Dezimalkonstante oder durch einen Verweis in Assemblersprache auf ein Feld, das als PL4 definiert ist, ersetzt werden. Der muss die Form 0HHMMSS + haben, wobei Folgendes gilt:

HH

Stellt Stunden von 00 bis 99 dar.

MM

Stellt Minuten von 00 bis 59 dar.

SS

Stellt Sekunden von 00 bis 59 dar.

Ein Argument *bezeichnung* wird für AMODE(64)-Programme nicht unterstützt.

Viele Befehle sind mit der Übertragung von Daten zwischen dem Anwendungsprogramm und CICS verbunden. In den meisten Fällen muss das Anwendungsprogramm die Länge der Daten, die übertragen werden sollen, angeben. Wenn ein Datenbereich jedoch als Quelle oder Ziel angegeben wird, ist es nicht erforderlich, die Länge explizit anzugeben, da der Befehlssprachenumsetzer eine Standardlänge generiert. Beispiel:

```
xxx DC CL8  
.  
.  
EXEC CICS ... LENGTH(L'xxx)
```

Einschränkungen für CICS-Befehle

Für alle CICS-Befehle, die auf Benutzerdaten zugreifen, gelten einige allgemeine Einschränkungen.

- Das Programm muss sich im primären Adressierungsmodus befinden, wenn es einen CICS-Service aufruft. Der Primäradressraum muss der Speicherbereich für die Ausgangsadresse sein. Alle Parameter, die an CICS übergeben werden, müssen sich im Primäradressraum befinden.
- Wenn Ihr Programm Zugriffsregister verwendet, behält CICS nur die Zugriffsregister 2 bis 13 bei, weil CICS-Code die Zugriffsregister 0, 1, 14 und 15 für z/OS-Makroaufrufe verwenden kann.

LENGTH-Optionen in CICS-Befehlen

In den Programmiersprachen COBOL, PL/I und Assembler nimmt der Umsetzer bestimmte Längen standardmäßig an, wenn die Umsetzeroptionen NOLENGTH nicht angegeben ist. Das bedeutet, dass sie in Programmen optional sind, die Datenbereiche angeben. In der Sprache C müssen alle Optionen LENGTH angegeben werden.

Wenn ein CICS-Befehl die Option LENGTH anbietet, wird diese durch einen Halbwort-Binärwert mit Vorzeichen dargestellt. Dies legt eine theoretische Obergrenze von 32 763 Byte für LENGTH fest. Nehmen Sie abhängig von Aspekten der Wiederherstellbarkeit, der Funktionsverlagerung und anderer Faktoren aus praktischen Gründen eine Begrenzung auf 24 KB an.

Diese empfohlene Begrenzung auf 24 KB gilt nicht für die Option FLENGTH in CICS-Befehlen (ausgenommen terminalbezogener Befehle **SEND** und **RECEIVE** wegen architekturbedingter Einschränkungen). Die Option FLENGTH wird in Befehlen verwendet, die sich unter anderem auf Container und Journale beziehen.

Für temporären Speicher, transiente Daten und Dateisteuerungsbefehle können die Dateidefinitionen selbst möglicherweise weitere Einschränkungen festlegen.

Option NOHANDLE

Mit der Option NOHANDLE können Sie bei jedem Befehl angeben, dass bei keiner der Bedingungen oder AID-Zeichen, die durch die Ausführung dieses Befehls hervorgerufen wurden, eine Aktion erfolgen soll.

Weitere Informationen zur Option NOHANDLE finden Sie unter [Optionen RESP und NOHANDLE](#).

Bei Verwendung der Programmiersprache C oder C++ wird die Option NOHANDLE in allen Befehlen impliziert.

Optionen RESP und RESP2

Die können die Option RESP mit jedem beliebigen Befehl verwenden, um zu testen, ob bei dessen Ausführung eine Bedingung ausgelöst wurde. Wenn Sie für bestimmte Befehle, bei denen eine Bedingung aus mehreren Gründen ausgelöst werden kann, bereits RESP angegeben haben, können Sie mithilfe der Option RESP2 genau bestimmen, warum eine Bedingung aufgetreten ist.

RESP(xxx)

xxx ist ein benutzerdefinierter Datenbereich für eine Vollwort-Binärzahl. Bei der Rückgabe durch den Befehl enthält er einen Wert, der einer möglicherweise aufgetretenen Bedingung oder aber einer normalen Rückgabe entspricht, also xxx=DFHRESP(NORMAL). Diesen Wert können Sie wie folgt mithilfe von DFHRESP testen:

```
EXEC CICS WRITEQ TS FROM(abc)
                        QUEUE(qname)
                        NOSUSPEND
                        RESP(xxx)
                        RESP2(yyy)
.
IF xxx=DFHRESP(NOSPACE) THEN ...
```

Dieses Format von DFHRESP gilt sowohl für COBOL als auch für PL/I.

Das folgende Beispiel zeigt einen ähnlichen Test in C:

```
switch (xxx) {
case DFHRESP(NORMAL) : break;
case DFHRESP(INVREQ) : Invreq_Cond();
                      break;
default              : Errors();
}
```

Das folgende Beispiel zeigt einen ähnlichen Test in Assemblersprache:

```
CLC    xxx,DFHRESP(NOSPACE)
```

Der Umsetzer ändert diesen Code in:

```
CLC    xxx,=F'18'
```

Da die Verwendung von RESP NOHANDLE impliziert, sollten Sie bei Verwendung von RESP mit dem Befehl RECEIVE Vorsicht walten lassen. NOHANDLE überschreibt sowohl den Befehl HANDLE AID als auch den Befehl HANDLE CONDITION mit dem Ergebnis, dass Antworten durch Funktionstasten ignoriert werden.

RESP2(yyy)

yyy ist ein benutzerdefinierter Datenbereich für eine Vollwort-Binärzahl. Bei der Rückgabe durch den Befehl enthält er einen Wert, der die Antwort auf bestimmte Befehle genauer qualifiziert. Anders als RESP-Werte besitzen RESP2-Werte keine zugeordneten symbolischen Namen und es gibt daher keine integrierte Funktion des Umsetzers, die DFHRESP entspricht, weshalb Sie den Vollwort-Binärwert selbst testen müssen.

Umgesetzter Code für CICS-Befehle

Anwendungsprogramme können in COBOL, C, PL/I oder Assemblersprache geschrieben sein und CICS-Befehle enthalten. CICS setzt diese Programme um und erstellt ein funktional entsprechendes Quellprogramm, in dem jeder Befehl dann in ein Aufrufmakro oder eine Aufrufanweisung in der Sprache des ursprünglichen Quellenprogramms umgesetzt ist.

COBOL-Konvertierungsausgabe

EXEC CICS-Befehle werden in Aufrufe der CICS-Schnittstelle DFHEI1 konvertiert.

Das folgende Beispiel zeigt, wie die EXEC-Anweisung

```
EXEC CICS RETURN TRANSID('fred')  
  COMMAREA(mycommarea) END-EXEC.
```

konvertiert wird:

```
Move length of mycommarea to dfhb0020  
Call 'DFHEI1' using by content  
  x'0e08e0000700001000f0f0f2f7404040'  
  by content 'fred' by reference mycommarea  
  by reference dfhb0020 end-call.
```

Copybook DFHEIBLC

Dieses neue Copybook ist eine Version des vorhandenen Copybooks DFHEIBLK in Kleinschreibung.

Ein Unterschied ist der Name der höchsten Ebene in DFHEIBLK:

```
01 EIBLK
```

In DFHEIBLC lautet der Name der höchsten Ebene hingegen wie folgt:

```
01 dfheiblk
```

Dies ist mit dem aktuellen vom Umsetzungsprogramm generierten Namen konsistent und entspricht außerdem der Regel, dass für CICS reservierte Wörter mit der Zeichenfolge 'DFH' beginnen sollten.

C-Konvertierungsausgabe

Bei einem C-Anwendungsprogramm wird jeder Befehl durch Anweisungen für eine erneute Zuordnung ersetzt, auf die eine Anweisung 'dfhexec' folgt, in der die Parameter übergeben werden.

PL/I-Konvertierungsausgabe

Bei einem PL/I-Anwendungsprogramm wird jeder Befehl immer durch eine Anweisung DO, eine Deklaration für einen generierten Eintragsnamen, eine Anweisung CALL und eine Anweisung END ersetzt. Die ENTRY-Deklaration stellt sicher, dass die geeigneten Konvertierungen für Argumentwerte stattfinden.

Wenn eine ON-Einheit in PL/I aus einem einzigen EXEC CICS-Befehl besteht, sollte der Befehl in einem BEGIN-Block angegeben sein. Beispiel:

```
ON ERROR BEGIN;  
  EXEC CICS RETURN;  
END;
```

Falls ein EXEC CICS-Befehl einem PL/I-Bedingungspräfix zugeordnet ist, sollte sich der Befehl ebenfalls in einem BEGIN-Block befinden. Beispiel:

```
(NOZERODIVIDE): BEGIN;  
  EXEC CICS GETMAIN  
  SET(zeigerverweis)  
  LENGTH(datenwert);  
END;
```

Falls OPTIONS(MAIN) angegeben ist, ändert das Umsetzungsprogramm die Parameterliste und fügt den EIB-Strukturzeiger als ersten Parameter ein. Ist OPTIONS(MAIN) nicht angegeben (soll das Programm also durch Linkbearbeitung mit dem Hauptmodul verbunden werden), wird die Parameterliste nicht geän-

dert; in diesem Fall muss der Anwendungsprogrammierer dafür sorgen, dass die EIB-Struktur im Programm mit Linkbearbeitung adressiert wird, falls ein Zugriff darauf erforderlich ist. In beiden Fällen fügt das Umsetzungsprogramm die Deklaration der EIB-Struktur ein, wenn ein Programm mit einer gültigen PL/I-Anweisung PROCEDURE beginnt.

Assemblerkonvertierungsausgabe

Der Aufruf eines CICS-Anwendungsprogramms in Assemblersprache befolgt Systemstandards.

Beim Einstieg in das Anwendungsprogramm enthalten die Register 1, 15, 14 und 13 die folgenden Adressen:

- Register 1 enthält die Adresse der Parameterliste. Diese Liste umfasst mindestens zwei Einträge:
 - Adresse des EXEC-Schnittstellenblocks (EIB).
 - Adresse des Kommunikationsbereichs (COMMAREA); falls es keinen Kommunikationsbereich gibt, lautet die Adresse X'00000000'.
- Register 15 enthält die Adresse des Einstiegspunkts.
- Register 14 enthält die Adresse des Rückgabepunkts.
- Register 13 enthält die Adresse des Sicherungsbereichs.

Alle anderen Register sind nicht definiert.

Makro DFHECALL

Wenn der CICS-Umsetzer für ein Anwendungsprogramm in Assemblersprache einen CICS-Befehl erkennt, wird jeder Befehl durch einen Aufruf des Makros DFHECALL ersetzt. Das Makro DFHECALL richtet die Befehlsparameter ein und ruft den ersten CICS-Befehlsprozessor auf, um den Befehl zu verarbeiten.

Dieses Makro wird in eine Systemstandardaufrufsequenz erweitert, die die Register 15, 14, 0 und 1 verwendet. Diese Register haben den folgenden Inhalt:

- Register 15 enthält die Adresse des Einstiegspunkts im EXEC-Schnittstellenprogramm.
- Register 14 enthält die Adresse des Rückgabepunkts in Ihrem Anwendungsprogramm.
- Register 0 ist nicht definiert.
- Register 1 enthält die Adresse der Parameterliste.

Der Einstiegspunkt, der in Register 15 enthalten ist, wird im EXEC-Schnittstellenprozessor aufgelöst, der mit Ihrem Anwendungsprogramm durch Linkbearbeitung verbunden sein muss. Für Anwendungen mit AMODE(24) und AMODE(31) ist dieser EXEC-Schnittstellenprozessor DFHEAI, für Anwendungen mit AMODE(64) ist dieser Prozessor DFHEAG.

Sie können den Exit über das Anwendungsprogramm durch einen Befehl EXEC CICS RETURN in Ihrem Quellenprogramm angeben. Alternativ können Sie das Makro DFHEIRET verwenden, das die Register wiederherstellt und die Steuerung an die Adresse in Register 14 zurückgibt. Der Umsetzer fügt das Makro DFHEIRET ohne Angabe von Parametern direkt vor der Anweisung END ein, sofern Sie nicht die Umsetzoption NOEPILOG angeben. Sie können dieses Makro verwenden, um von einem Programm der höchsten Ebene zurückzukehren, jedoch ist es nicht empfehlenswert, dies von einem Programm einer niedrigeren Ebene zu tun.

Während der Assemblierung erstellt das Makro DFHECALL eine Argumentliste im dynamischen Speicher, sodass das Anwendungsprogramm wiedereintrittsfähig (REENTRANT) ist. Anschließend ruft das Makro das EXEC-Schnittstellenprogramm DFHEIP für Anwendungen mit AMODE(24) oder AMODE(31) bzw. DFHEIG für Anwendungen mit AMODE(64) auf. Diese Programme müssen auch den Systemstandards entsprechen, wie dies zuvor beschrieben wurde.

Für Anwendungen mit AMODE(64) gilt, dass die Anwendung und der erste Befehlsprozessor zwar im 64-Bit-Adressierungsmodus ausgeführt werden, die Parameter jedoch, die das Makro DFHECALL einrichtet und an den ersten Befehlsprozessor übergibt, 31-Bit-Adressen enthalten. Daher muss der Speicher, in dem die Aufrufparameter erstellt werden (DFHEISTG-Speicher), 31-Bit-Speicher (über 16 MB, jedoch unter 2 GB) sein.

Neben dem Aufruf des Makros DFHECALL fügt der Umsetzer auch die folgenden Makros in Ihr Quellprogramm ein:

DFHEIGBL

Dieses Makro legt globale Einstellungen fest, wenn Sie EXEC DLI entweder in einem Stapelanwendungsprogramm oder in einem Online-CICS-Anwendungsprogramm verwenden. Wenn in DFHEIGBL für DFHEIDL der Wert 1 angegeben ist, bedeutet dies, dass das Programm EXEC-DLI-Befehle enthält. Wenn DFHEIDB auf 1 gesetzt ist, bedeutet dies, dass das Programm ein Stapel-DL/I-Programm ist. Wenn Sie DL/I nicht verwenden, ist es herauskommentiert und auf den Wert 0 gesetzt.

DFHEIENT

Dieses Makro wird nach der ersten Anweisung CSECT oder START eingefügt. Es führt Prologcode aus, um Arbeitsspeicher zur Aufnahme von Benutzervariablen sowie zur Verwendung durch CICS zuzuordnen:

- Es speichert Register.
- Es ruft eine Anfangszuordnung des Speichers ab, der durch DFHEISTG definiert wird.
- Es richtet ein Basisadressregister (Standardregister 3) ein.
- Es richtet ein Register für dynamischen Speicher (Standardregister 13) ein.
- Es richtet ein Register zur Adressierung des EIB (Standardregister 11) ein.

DFHEIRET

Dieses Makro führt Epilogcode aus, um den Arbeitsspeicher des Anwendungsprogramms freizugeben:

- Es stellt Register wieder her.

DFHEIRET RCREG=nn, wobei nn (beliebige Registernummer außer 13) den Rückgabecode enthält, der in das Register 15 nach Wiederherstellung der Register eingefügt werden soll.

- Es gibt die Steuerung an die Adresse in Register 14 zurück.

DFHEISTG und DFHEIEND

Diese Makros definieren dynamischen Speicher:

- Sie definieren den Speicher, der für die Parameterliste erforderlich ist.
- Sie definieren einen Sicherungsbereich.

Weitere Einzelheiten zu diesen Makros mit AMODE(64)-Anwendungen finden Sie unter [Assemblerschnittstelle EXEC CICS codieren](#).

Es wird außerdem ein Copybook DFHEIBLK, das einen DSECT enthält, der den EIB beschreibt, automatisch eingeschlossen.

Das Programm muss eine Anweisung END haben, weil der Umsetzer die Standardmakros andernfalls nicht einfügt. Darüber hinaus müssen CSECT oder START und END in Großschreibung angegeben sein, damit der Umsetzer sie erkennt.

Das Beispiel in [Abbildung 1 auf Seite 15](#) zeigt ein einfaches Assemblersprachenprogramm, das den BMS-Befehl SEND MAP verwendet, um eine Maske an ein Terminal zu senden. Darauf folgt die Ausgabe, nachdem das Programm INSTRUCT umgesetzt wurde.

Quellenprogramm

```
INSTRUCT CSECT
  EXEC CICS SEND MAP('DFH$AGA') MAPONLY ERASE
END
```

Dieses Programm wird wie folgt umgesetzt:

```
          DFHEIGBL ,          INSERTED BY TRANSLATOR
INSTRUCT CSECT
  DFHEIENT          INSERTED BY TRANSLATOR
*  EXEC CICS SEND MAP('DFH$AGA') MAPONLY ERASE
  DFHECALL =X'1804C00008000000000046204000020',
          (CHA7,=CL7'DFH$AGA*'),(____RF,DFHEIV00)
  DFHEIRET          INSERTED BY TRANSLATOR
  DFHEISTG          INSERTED BY TRANSLATOR
  DFHEIEND          INSERTED BY TRANSLATOR
END
```

Abbildung 1. Quellenprogramm und umgesetzter Code für einen CICS-Befehl

Kapitel 2. Zusammenfassung der CICS-Befehle

Die **EXEC CICS**-Befehle sind nach der Funktion kategorisiert, die sie ausführen.

EIB-Antwortcodes finden Sie unter [Antwortcodes von EXEC CICS-Befehlen](#).

EIB-Funktionscodes finden Sie unter [Funktionscodes von EXEC CICS-Befehlen](#).

Unterstützung für abnormale Beendigung (ABEND)

- [ABEND](#)
- [HANDLE ABEND](#)

APPC-Basisdatenaustausch (Basisdialog)

- [GDS ALLOCATE](#)
- [GDS ASSIGN](#)
- [GDS CONNECT PROCESS](#)
- [GDS EXTRACT ATTRIBUTES](#)
- [GDS EXTRACT PROCESS](#)
- [GDS FREE](#)
- [GDS ISSUE ABEND](#)
- [GDS ISSUE CONFIRMATION](#)
- [GDS ISSUE ERROR](#)
- [GDS ISSUE PREPARE](#)
- [GDS ISSUE SIGNAL](#)
- [GDS RECEIVE](#)
- [GDS SEND](#)
- [GDS WAIT](#)

Formatfreier APPC-Datenaustausch (formatfreier APPC-Dialog)

- [ALLOCATE \(APPC\)](#)
- [CONNECT PROCESS](#)
- [CONVERSE \(APPC\)](#)
- [EXTRACT ATTRIBUTES \(APPC\)](#)
- [EXTRACT PROCESS](#)
- [FREE \(APPC\)](#)
- [ISSUE ABEND](#)
- [ISSUE CONFIRMATION](#)
- [ISSUE ERROR](#)
- [ISSUE PREPARE](#)
- [ISSUE SIGNAL \(APPC\)](#)
- [RECEIVE \(APPC\)](#)

- [SEND \(APPC\)](#)
- [WAIT CONVID \(APPC\)](#)

Asynchrone Services

- [FETCH ANY](#)
- [FETCH CHILD](#)
- [FREE CHILD](#)
- [RUN TRANSID](#)

Authentifizierung

- [CHANGE PASSWORD](#)
- „CHANGE PHRASE“ auf Seite 71
- [SIGNOFF](#)
- [SIGNON](#)
- [SIGNON TOKEN](#)
- [VERIFY PASSWORD](#)
- [VERIFY PHRASE](#)
- [VERIFY TOKEN](#)

Stapeldatenaustausch (BDI - Batch Data Interchange)

- [ISSUE ABORT](#)
- [ISSUE ADD](#)
- [ISSUE END](#)
- [ISSUE ERASE](#)
- [ISSUE NOTE](#)
- [ISSUE QUERY](#)
- [ISSUE RECEIVE](#)
- [ISSUE REPLACE](#)
- [ISSUE SEND](#)
- [ISSUE WAIT](#)

BMS

- [PURGE MESSAGE](#)
- [RECEIVE MAP](#)
- [RECEIVE MAP MAPPINGDEV](#)
- [RECEIVE PARTN](#)
- [ROUTE](#)
- [SEND CONTROL](#)
- [SEND MAP](#)
- [SEND MAP MAPPINGDEV](#)

- [SEND PAGE](#)
- [SEND PARTNSET](#)
- [SEND TEXT](#)
- [SEND TEXT MAPPED](#)
- [SEND TEXTNOEDIT](#)

Integrierte Funktionen

- [BIF DEEDIT](#)
- [BIF DIGEST](#)

CICS Business Transaction Services (BTS)

- [ACQUIRE](#)
- [ADD SUBEVENT](#)
- [CANCEL](#)
- [„CANCEL \(BTS\)“ auf Seite 69](#)
- [CHECK ACQPROCESS](#)
- [CHECK ACTIVITY](#)
- [CHECK TIMER](#)
- [DEFINE ACTIVITY](#)
- [DEFINE COMPOSITE EVENT](#)
- [DEFINE INPUT EVENT](#)
- [DEFINE PROCESS](#)
- [DEFINE TIMER](#)
- [DELETE ACTIVITY](#)
- [DELETE CONTAINER \(BTS\)](#)
- [DELETE EVENT](#)
- [DELETE TIMER](#)
- [ENDBROWSE ACTIVITY](#)
- [„ENDBROWSE CONTAINER \(BTS\)“ auf Seite 162](#)
- [ENDBROWSE EVENT](#)
- [ENDBROWSE PROCESS](#)
- [FORCE TIMER](#)
- [GET CONTAINER \(BTS\)](#)
- [GETNEXT ACTIVITY](#)
- [„GETNEXT CONTAINER \(BTS\)“ auf Seite 253](#)
- [GETNEXT EVENT](#)
- [GETNEXT PROCESS](#)
- [INQUIRE ACTIVITYID](#)
- [INQUIRE CONTAINER](#)
- [INQUIRE EVENT](#)
- [INQUIRE PROCESS](#)
- [INQUIRE TIMER](#)

- [LINK ACQPROCESS](#)
- [LINK ACTIVITY](#)
- [MOVE CONTAINER \(BTS\)](#)
- [PUT CONTAINER \(BTS\)](#)
- [REMOVE SUBEVENT](#)
- [RESET ACQPROCESS](#)
- [RESET ACTIVITY](#)
- [RESUME](#)
- [RETRIEVE REATTACH EVENT](#)
- [RETRIEVE SUBEVENT](#)
- [RUN](#)
- [STARTBROWSE ACTIVITY](#)
- [„STARTBROWSE CONTAINER \(BTS\)” auf Seite 591](#)
- [STARTBROWSE EVENT](#)
- [STARTBROWSE PROCESS](#)
- [SUSPEND \(BTS\)](#)
- [TEST EVENT](#)

Kanalbefehle

- [DELETE CHANNEL](#)
- [DELETE CONTAINER \(CHANNEL\)](#)
- [„ENDBROWSE CONTAINER \(CHANNEL\)” auf Seite 162](#)
- [GET CONTAINER \(CHANNEL\)](#)
- [GET64 CONTAINER](#)
- [„GETNEXT CONTAINER \(CHANNEL\)” auf Seite 254](#)
- [MOVE CONTAINER \(CHANNEL\)](#)
- [PUT CONTAINER \(CHANNEL\)](#)
- [PUT64 CONTAINER](#)
- [QUERY CHANNEL](#)
- [„STARTBROWSE CONTAINER \(CHANNEL\)” auf Seite 593](#)
- [START CHANNEL](#)

Konsolenunterstützung

- [WRITE OPERATOR](#)

Diagnoseservices

- [DUMP TRANSACTION](#)
- [ENTER TRACENUM](#)

Dokumentenservices

- DOCUMENT CREATE
- DOCUMENT DELETE
- DOCUMENT INSERT
- DOCUMENT RETRIEVE
- DOCUMENT SET

Umgebungsservices

- ADDRESS
- ADDRESS SET
- ASSIGN

Ereignisverarbeitung

- SIGNAL EVENT

Unterstützung für Ausnahmebedingungen

- HANDLE CONDITION
- IGNORE CONDITION
- POP HANDLE
- PUSH HANDLE

Dateisteuerungsservices

- DELETE
- ENDBR
- READ
- READNEXT
- READPREV
- RESETBR
- REWRITE
- STARTBR
- UNLOCK
- WRITE

Intervallsteuerungsservices

- ASKTIME
- CANCEL
- DELAY
- FORMATTIME
- POST
- RETRIEVE

- [START](#)
- [WAIT EVENT](#)

Journalführung

- [WAIT JOURNALNAME](#)
- [WAIT JOURNALNUM](#)
- [WRITE JOURNALNAME](#)
- [WRITE JOURNALNUM](#)

Überwachung

- [MONITOR](#)

Server für benannte Zähler

- [DEFINE COUNTER](#) und [DEFINE DCOUNTER](#)
- [DELETE COUNTER](#) und [DELETE DCOUNTER](#)
- [GET COUNTER](#) und [GET DCOUNTER](#)
- [QUERY COUNTER](#) und [QUERY DCOUNTER](#)
- [REWIND COUNTER](#) und [REWIND DCOUNTER](#)
- [UPDATE COUNTER](#) und [UPDATE DCOUNTER](#)

Programmsteuerung

- „[INVOKE APPLICATION](#)“ auf Seite 278
- [LINK](#)
- [LOAD](#)
- [RELEASE](#)
- [RETURN](#)
- [XCTL](#)

Zeitplanungsservices

- [START ATTACH](#)
- [START BREXIT](#)

Sicherheitsservices

- [REQUEST PASSTICKET](#)
- [QUERY SECURITY](#)

Spool Interface (JES)

- [SPOOLCLOSE](#)
- [SPOOLOPEN INPUT](#)
- [SPOOLOPEN OUTPUT](#)
- [SPOOLREAD](#)
- [SPOOLWRITE](#)

Speichersteuerung

- [FREEMAIN](#)
- [FREEMAIN64](#)
- [GETMAIN](#)
- [GETMAIN64](#)

Synchronisationspunkt

- [SYNCPOINT](#)
- [SYNCPOINT ROLLBACK](#)

Tasksteuerung

- [CHANGE TASK](#)
- [DEQ](#)
- [ENQ](#)
- [SUSPEND](#)
- [WAIT EXTERNAL](#)
- [WAITCICS](#)

TCP/IP-Services

- [EXTRACT CERTIFICATE](#)
- [EXTRACT TCPIP](#)

Steuerung des temporären Speichers

- [DELETEQ TS](#)
- [READQ TS](#)
- [WRITEQ TS](#)

Terminalsteuerung

- [ALLOCATE \(LUTYPE6.1\)](#)
- [ALLOCATE \(MRO\)](#)
- [BUILD ATTACH \(LUTYPE6.1\)](#)
- [BUILD ATTACH \(MRO\)](#)

- [CONVERSE \(APPC\)](#)
- [„CONVERSE \(Standard\)” auf Seite 84](#)
- [CONVERSE \(LUTYPE2/LUTYPE3\)](#)
- [CONVERSE \(LUTYPE4\)](#)
- [CONVERSE \(LUTYPE6.1\)](#)
- [CONVERSE \(MRO\)](#)
- [„CONVERSE \(Nicht-z/OS Communications Server-Standard\)” auf Seite 98](#)
- [„CONVERSE: Nicht-z/OS Communications Server-Optionen” auf Seite 99](#)
- [CONVERSE \(SCS\)](#)
- [„CONVERSE: z/OS Communications Server-Optionen” auf Seite 93](#)
- [CONVERSE \(2260\)](#)
- [CONVERSE \(3270 logisch\)](#)
- [CONVERSE \(3600-3601\)](#)
- [CONVERSE \(3600-3614\)](#)
- [CONVERSE \(3650-Interpreter\)](#)
- [CONVERSE \(3650-3270\)](#)
- [CONVERSE \(3650-3653\)](#)
- [CONVERSE \(3650-3680\)](#)
- [CONVERSE \(3767\)](#)
- [CONVERSE \(3770\)](#)
- [CONVERSE \(3790 mit vollem Funktionsumfang oder Anfrage\)](#)
- [CONVERSE \(3790- bzw. 3270-Anzeige\)](#)
- [EXTRACT ATTACH \(LUTYPE6.1\)](#)
- [EXTRACT ATTACH \(MRO\)](#)
- [EXTRACT ATTRIBUTES \(MRO\)](#)
- [EXTRACT LOGONMSG](#)
- [EXTRACT TCT](#)
- [FREE \(LUTYPE6.1\)](#)
- [FREE](#)
- [FREE \(MRO\)](#)
- [HANDLE AID](#)
- [ISSUE COPY \(3270 logisch\)](#)
- [„ISSUE DISCONNECT \(LUTYPE6.1\)” auf Seite 296](#)
- [ISSUE DISCONNECT](#)
- [ISSUE ENDFILE](#)
- [ISSUE ENDOUTPUT](#)
- [ISSUE EODS](#)
- [ISSUE ERASEAUP](#)
- [ISSUE LOAD](#)
- [ISSUE PASS](#)
- [ISSUE PRINT](#)
- [ISSUE RESET](#)
- [ISSUE SIGNAL \(LUTYPE6.1\)](#)

- [POINT](#)
- [RECEIVE \(APPC\)](#)
- [RECEIVE \(LUTYPE2/LUTYPE3\)](#)
- [RECEIVE \(LUTYPE4\)](#)
- [RECEIVE \(LUTYPE6.1\)](#)
- [RECEIVE \(MRO\)](#)
- [„RECEIVE \(Standardformat ohne z/OS Communications Server\)” auf Seite 424](#)
- [„RECEIVE: Optionen ohne z/OS Communications Server ” auf Seite 429](#)
- [„RECEIVE \(Standardformat für z/OS Communications Server\)” auf Seite 414](#)
- [„RECEIVE \(Standardformat für z/OS Communications Server\)” auf Seite 414](#)
- [RECEIVE \(2260\)](#)
- [RECEIVE \(2980\)](#)
- [RECEIVE \(3270 logisch\)](#)
- [„RECEIVE \(3600-Pipeline\)” auf Seite 417](#)
- [RECEIVE \(3600-3601\)](#)
- [RECEIVE \(3600-3614\)](#)
- [RECEIVE \(3650\)](#)
- [RECEIVE \(3767\)](#)
- [RECEIVE \(3770\)](#)
- [RECEIVE \(3790 mit vollem Funktionsumfang oder Anfrage\)](#)
- [RECEIVE \(3790- bzw. 3270-Anzeige\)](#)
- [SEND \(APPC\)](#)
- [SEND \(LUTYPE2/LUTYPE3\)](#)
- [SEND \(LUTYPE4\)](#)
- [SEND \(LUTYPE6.1\)](#)
- [SEND \(MRO\)](#)
- [„SEND \(Standardformat ohne z/OS Communications Server\)” auf Seite 496](#)
- [„SEND: Optionen für Nicht-z/OS Communications Server” auf Seite 497](#)
- [SEND \(SCS\)](#)
- [„SEND \(Standardformat für z/OS Communications Server\)” auf Seite 482](#)
- [„SEND: Optionen für z/OS Communications Server” auf Seite 492](#)
- [SEND \(2260\)](#)
- [SEND \(2980\)](#)
- [SEND \(3270 logisch\)](#)
- [SEND \(3600-Pipeline\)](#)
- [SEND \(3600-3601\)](#)
- [SEND \(3600-3614\)](#)
- [SEND \(3650-Interpreter\)](#)
- [SEND \(3650-3270\)](#)
- [SEND \(3650-3653\)](#)
- [SEND \(3650-3680\)](#)
- [SEND \(3767\)](#)
- [SEND \(3770\)](#)

- [SEND \(3790 mit vollem Funktionsumfang oder Anfrage\)](#)
- [SEND \(3790 SCS\)](#)
- [SEND \(3790- bzw. 3270-Anzeige\)](#)
- [SEND \(3790- bzw. 3270-Drucker\)](#)
- [WAIT SIGNAL](#)
- [WAIT TERMINAL](#)

Transiente Daten

- [DELETEQ TD](#)
- [READQ TD](#)
- [WRITEQ TD](#)

Webunterstützung

- [CONVERTTIME](#)
- [EXTRACT WEB](#)
- [WEB CLOSE](#)
- [WEB CONVERSE](#)
- [WEB ENDBROWSE FORMFIELD](#)
- [WEB ENDBROWSE HTTPHEADER](#)
- [„WEB ENDBROWSE QUERYPARM“ auf Seite 660](#)
- [WEB EXTRACT](#)
- [WEB OPEN](#)
- [WEB PARSE URL](#)
- [WEB READ FORMFIELD](#)
- [WEB READ HTTPHEADER](#)
- [„WEB READ QUERYPARM“ auf Seite 678](#)
- [WEB READNEXT FORMFIELD](#)
- [WEB READNEXT HTTPHEADER](#)
- [„WEB READNEXT QUERYPARM“ auf Seite 683](#)
- [WEB RECEIVE \(Server\)](#)
- [WEB RECEIVE \(Client\)](#)
- [WEB RETRIEVE](#)
- [WEB SEND \(Server\)](#)
- [WEB SEND \(Client\)](#)
- [WEB STARTBROWSE FORMFIELD](#)
- [WEB STARTBROWSE HTTPHEADER](#)
- [„WEB STARTBROWSE QUERYPARM“ auf Seite 722](#)
- [WEB WRITE HTTPHEADER](#)

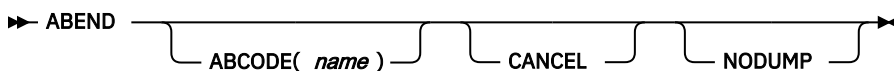
Web-Services

- [INVOKE SERVICE](#)
- [INVOKE WEBSERVICE](#)
- [SOAPFAULT ADD](#)
- [SOAPFAULT CREATE](#)
- [SOAPFAULT DELETE](#)
- [„TRANSFORM DATATOJSON“ auf Seite 602](#)
- [TRANSFORM DATATOXML](#)
- [„TRANSFORM JSONTODATA“ auf Seite 606](#)
- [TRANSFORM XMLTODATA](#)
- [WSACONTEXT BUILD](#)
- [WSACONTEXT DELETE](#)
- [WSACONTEXT GET](#)
- [WSAEPR CREATE](#)

ABEND

Führt eine abnormale Beendigung (Abbruch) einer Task aus.

ABEND



Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl **ABEND** beendet eine Task abnormal.

CICS gibt den Hauptspeicher frei, der der beendeten Task zugeordnet ist. Optional können Sie einen Transaktionsspeicherauszug dieses Speichers abrufen.

Durch den Aufruf des Befehls **ABEND** wird die aktuelle Transaktion abnormal beendet. Language Environment wird informiert, dass ein Abbruch erfolgt ist und die folgende Nachricht wird an CEEMSG geschrieben, an die sich ein Speicherauszugsbericht anschließt:

```
CEE3250C The system or user abend XXXX was issued
```

XXXX ist der Transaktionsspeicherauszugscode, der in der Option ABCODE angegeben wurde. Language Environment versucht, auf Registeradressen zuzugreifen, um den referenzierten Speicher für den Speicherauszugsbericht zu extrahieren, der in CEEMSG geschrieben wird. Wenn Language Environment keinen Zugriff auf den Speicher hat, der von diesen Registern adressiert wird, kann eine abnormale Beendigung OC4 auftreten. Sie können abnormale Beendigungen OC4 vermeiden, indem Sie die Language Environment-Laufzeitoption TERMTHDACT auf QUIET, MSG oder UAONLY setzen. Weitere Informationen finden Sie unter [TERMTHDACT in z/OS Language Environment Customization](#).

Optionen

ABCODE(name)

Diese Option gibt an, dass ein Speicherauszug des Hauptspeichers, der zu der beendeten Task gehört, erstellt wird. Der Wert von ABCODE wird als Transaktionsspeicherauszugscode verwendet, um den Speicherauszug zu identifizieren. Für ABCODE gelten die gleichen Formatregeln wie für DUMPCODE. In den Informationen zum Befehl „DUMP TRANSACTION“ auf Seite 154 werden die Formatregeln beschrieben, die für DUMPCODE gelten. Wenn diese Regeln nicht befolgt werden, erstellt der Befehl ABEND keinen Speicherauszug.

Beginnen Sie den Namen nicht mit dem Buchstaben A, da dieser für CICS selbst reserviert ist.

Anmerkung: Wenn die Option ABCODE nicht angegeben wird, ist der Effekt derselbe wie bei Angabe der Option NODUMP.

CANCEL

Diese Option gibt an, dass Exits, die durch Befehle **HANDLE ABEND** eingerichtet wurden, ignoriert werden. Ein Befehl **ABEND CANCEL** bricht alle Exits auf jeder Ebene in der Task ab und beendet die Task abnormal. Wenn die PL/I-Ausführungszeitoption **STAE** angegeben ist, wird von PL/I ein Exit für abnormale Beendigung eingerichtet. Dieser Exit wird durch die Option **CANCEL** entfernt.

NODUMP

Diese Option gibt an, dass die abnormale Beendigung ohne Erstellung eines Speicherauszugs erfolgt. Für Programme, deren Links mit der Language Environment-Bibliothek SCEELKED bearbeitet wurden, wird bei Angabe der Option NODUMP unabhängig von den Einstellungen in der Transaktionsspeicherauszugstabelle nie ein Speicherauszug erstellt. Für Programme, deren Links nicht unter Verwendung von Language Environment bearbeitet wurden, wird die Option NODUMP ignoriert, wenn die Transaktionsspeicherauszugstabelle bereits einen Eintrag für den Abbruchcode enthält oder wenn die abnormale Beendigung bei der Initialisierung oder Beendigung der Language Environment-Ausführungseinheit erfolgt.

Beispiele

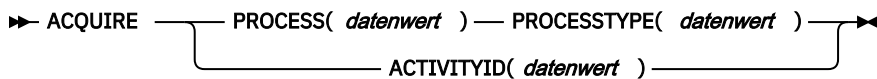
Das folgende Beispiel zeigt, wie eine Task abnormal beendet wird.

EXEC CICS ABEND ABCODE('BCDE')

ACQUIRE

Fordert Zugriff auf eine BTS-Aktivität von außerhalb des Prozesses an, in dem sie enthalten ist.

ACQUIRE PROCESS



Bedingungen: ACTIVITYBUSY, ACTIVITYERR, INVREQ, IOERR, LOCKED, NOTAUTH, PROCESSBUSY, PROCESSERR

Beschreibung

Mit dem Befehl ACQUIRE kann ein Programm, das außerhalb eines bestimmten BTS-Prozesses ausgeführt wird, auf eine Aktivität innerhalb des Prozesses zugreifen. Dies gibt dem Programm die folgenden Möglichkeiten:

- Es kann Daten in den Datencontainern der Aktivität lesen und schreiben.
- Es kann verschiedene Befehle wie beispielsweise RUN und LINK für die Aktivität ausgeben.¹

Eine Aktivität, auf die ein Programm mit einem Befehl ACQUIRE zugreift, wird als **angeforderte Aktivität** bezeichnet. Ein Programm kann nur eine einzige Aktivität pro Arbeitseinheit anfordern. Die Aktivität bleibt bis zum nächsten Synchronisationspunkt angefordert.

Der Befehl ACQUIRE ACTIVITYID fordert die angegebene untergeordnete Aktivität (keine Stammaktivität) an.

Der Befehl ACQUIRE PROCESS fordert die Stammaktivität des angegebenen Prozesses an.

Anmerkung: Wenn ein Programm einen Prozess definiert, erhält es automatisch Zugriff auf die Stammaktivität des Prozesses. (Dies versetzt das definierende Programm in die Lage, auf die Prozesscontainer und

¹ Falls es sich bei der angeforderten Aktivität um eine Stammaktivität handelt, kann das Programm Befehle für den Prozess ausführen.

die Container der Stammaktivität zuzugreifen, bevor der Prozess ausgeführt wird.) Wenn ein Programm den Zugriff auf eine Stammaktivität *entweder* durch einen Befehl DEFINE PROCESS *oder* durch einen Befehl ACQUIRE PROCESS erhält, wird der Prozess als **angeforderter Prozess** bezeichnet.

Regeln

1. Ein Programm kann in einer Arbeitseinheit nur eine einzige Aktivität anfordern. Die Aktivität bleibt bis zum nächsten Synchronisationspunkt angefordert. Dies bedeutet beispielsweise Folgendes:
 - Ein Programm kann in derselben Arbeitseinheit nicht sowohl einen Befehl DEFINE PROCESS als auch einen Befehl ACQUIRE PROCESS ausgeben.
 - Ein Programm kann in derselben Arbeitseinheit nicht sowohl einen Befehl ACQUIRE PROCESS als auch einen Befehl ACQUIRE ACTIVITYID ausgeben. Es kann also *entweder* eine untergeordnete Aktivität *oder* eine Stammaktivität anfordern, jedoch nicht beides.
2. Falls das Programm als Aktivierung einer Aktivität ausgeführt wird, ist Folgendes nicht möglich:
 - Das Programm kann eine Aktivität nicht in demselben Prozess anfordern, in dem es sich selbst befindet. Es kann beispielsweise keinen Befehl ACQUIRE PROCESS für den aktuellen Prozess ausgeben.
 - Das Programm kann keinen Befehl LINK verwenden, um die angeforderte Aktivität zu aktivieren.
3. Auf den Prozess einer angeforderten Aktivität kann auf dieselbe Weise zugegriffen werden, mit der die Aktivität selbst auf den Prozess zugreifen kann. Falls es sich bei der angeforderten Aktivität um eine untergeordnete Aktivität handelt, gilt daher Folgendes:
 - Die Container ihres Prozesses können gelesen, jedoch nicht aktualisiert werden.
 - Der Prozess kann nicht Gegenstand von Befehlen wie RUN, LINK, SUSPEND, RESUME oder RESET sein, die den Prozess oder seine Stammaktivität direkt bearbeiten.

Falls es sich bei der angeforderten Aktivität um eine Stammaktivität handelt, gilt umgekehrt Folgendes:

- Die Container ihres Prozesses können sowohl gelesen als auch aktualisiert werden.
- Der Prozess kann Gegenstand von Befehlen wie RUN, LINK, SUSPEND, RESUME oder RESET sein. Das Schlüsselwort ACQPROCESS im Befehl gibt als betreffenden Prozess denjenigen Prozess an, den das Programm, das den Befehl ausgibt, in der aktuellen Arbeitseinheit angefordert hat.

Optionen

ACTIVITYID(datenwert)

Diese Option gibt die ID (1 bis 52 Zeichen) der anzufordernden untergeordneten Aktivität an.

PROCESS(datenwert)

Diese Option gibt den Namen (1 bis 36 Zeichen) des Prozesses an, dessen Stammaktivität angefordert werden soll.

PROCESSTYPE(datenwert)

Diese Option gibt den Prozesstyp (1 bis 8 Zeichen) des Prozesses an, dessen Stammaktivität angefordert werden soll.

Bedingungen

107 ACTIVITYBUSY

RESP2-Werte:

19

Die Anforderung hat das Zeitlimit überschritten. Es kann sein, dass die Beendigung einer anderen Task, die diesen Aktivitätsdatensatz verwendet, verhindert wurde.

109 ACTIVITYERR

RESP2-Werte:

8

Die in der Option ACTIVITYID angegebene Aktivität wurde nicht gefunden.

16 INVREQ

RESP2-Werte:

22

Die Arbeitseinheit, die den Befehl ACQUIRE ausgegeben hat, hat bereits eine Aktivität angefordert. Eine Arbeitseinheit kann jeweils nur eine einzige Aktivität anfordern.

17 IOERR

RESP2-Werte:

29

Die Repository-Datei ist nicht verfügbar.

30

Für die Repository-Datei trat ein Ein-/Ausgabefehler auf.

100 LOCKED

Die Anforderung kann nicht ausgeführt werden, weil eine Sperre für den relevanten Datensatz in der Repository-Datei besteht.

70 NOTAUTH

RESP2-Werte:

101

Der Benutzer, der der ausgebenden Task zugeordnet ist, ist nicht zum Zugriff auf die Datei berechtigt, die zur BTS-Repository-Datei gehört, in der Details des Prozesses gespeichert sind.

106 PROCESSBUSY

RESP2-Werte:

13

Die Anforderung hat das Zeitlimit überschritten. Es kann sein, dass die Beendigung einer anderen Task, die diesen Prozessdatensatz verwendet, verhindert wurde.

108 PROCESSERR

RESP2-Werte:

5

Der in der Option PROCESS angegebene Prozess wurde nicht gefunden.

9

Der in der Option PROCESSTYPE angegebene Prozesstyp wurde nicht gefunden.

Verwendungsbeispiele

Mit dem Befehl ACQUIRE ACTIVITYID können benutzerbezogene Aktivitäten implementiert werden. Bei ihrer erstmaligen Aktivierung kann eine Aktivität beispielsweise Folgendes ausführen:

1. Sie kann ein Eingabeereignis definieren, um eine bestimmte Benutzerinteraktion darzustellen.
2. Sie kann einen Befehl ASSIGN ausgeben, um die ID ihrer eigenen Aktivitätsinstanz abzurufen.
3. Sie kann das Eingabeereignis und die Aktivitäts-ID in einer Datenbank speichern.
4. Sie kann ohne Abschluss eine Rückgabe vornehmen.

Wenn ein Benutzer später zur Verarbeitung der Task, die durch die Aktivität dargestellt wird, bereit ist, startet der Benutzer eine Transaktion. Diese außerhalb des BTS-Prozesses ausgeführte Transaktion führt Folgendes durch:

1. Sie ruft das Eingabeereignis und die Aktivitäts-ID aus der Datenbank ab.
2. Sie fordert mit dem Befehl ACQUIRE ACTIVITYID den Zugriff auf die Aktivität an.
3. Sie legt die Informationen, die zur Ausführung der Aktivität erforderlich sind, in einem Eingabedatencontainer ab und führt die Aktivität aus. Die Option INPUTEVENT des Befehls RUN teilt der Aktivität den Grund für ihre Aktivierung mit.

Mit dem Befehl ACQUIRE PROCESS kann eine Client/Server-Verarbeitung implementiert werden. Ein Clientprogramm kann beispielsweise die Befehle DEFINE PROCESS und RUN verwenden, um einen Ser-

verprozess zu erstellen und auszuführen, der eine bestimmte Aufgabe erledigt, eines oder mehrere Eingabeereignisse definiert und ohne Abschluss zurückgegeben wird. Der Client gibt einen Synchronisationspunkt aus oder nimmt eine Rückgabe vor. Um denselben Serverprozess erneut auszuführen, verwendet der Client die Befehle ACQUIRE PROCESS und RUN.

ADD SUBEVENT

Fügt ein Unterereignis zu einem zusammengesetzten BTS-Ereignis hinzu.

ADD SUBEVENT

➡ ADD — SUBEVENT(*datenwert*) — EVENT(*datenwert*) ➡

Bedingungen: EVENTERR, INVREQ

Beschreibung

Der Befehl ADD SUBEVENT fügt ein Unterereignis zu einem zusammengesetzten BTS-Ereignis hinzu. Für das Unterereignis gilt Folgendes:

- Es muss ein atomares (also nicht zusammengesetztes) Ereignis sein.
- Es kann kein Systemereignis sein.
- Es darf gegenwärtig nicht Teil des zusammengesetzten Ereignisses sein.
- Es kann kein Eingabeereignis sein, falls das Prädikat des zusammengesetzten Ereignisses den booleschen Operator AND verwendet.

Durch das Hinzufügen eines Unterereignisses wird das Prädikat des zusammengesetzten Ereignisses erneut ausgewertet.

Optionen

EVENT(*datenwert*)

Diese Option gibt den Namen (1 bis 16 Zeichen) des zusammengesetzten Ereignisses an. Dieser muss zuvor mit dem Befehl DEFINE COMPOSITE EVENT für die aktuelle Aktivität definiert worden sein.

SUBEVENT(*datenwert*)

Diese Option gibt den Namen (1 bis 16 Zeichen) des atomaren Ereignisses an, das als Unterereignis zum zusammengesetzten Ereignis hinzugefügt werden soll. Das Unterereignis muss zuvor mit einem der folgenden Befehle für die aktuelle Aktivität definiert worden sein:

- DEFINE ACTIVITY
- DEFINE INPUT EVENT
- DEFINE TIMER

Für das Unterereignis gilt Folgendes:

- Es darf gegenwärtig nicht Teil des zusammengesetzten Ereignisses sein.
- Es kann kein Eingabeereignis sein, falls das Prädikat des zusammengesetzten Ereignisses den booleschen Operator AND verwendet.

Bedingungen

111 EVENTERR

RESP2-Werte:

4

Das in der Option EVENT angegebene Ereignis wird durch BTS nicht erkannt.

5

Das in der Option SUBEVENT angegebene Unterereignis wird durch BTS nicht erkannt.

16 INVREQ

RESP2-Werte:

1

Der Befehl wurde außerhalb des Geltungsbereichs einer Aktivität ausgegeben.

2

Das in der Option EVENT angegebene Ereignis ist ungültig, weil es kein zusammengesetztes Ereignis ist.

3

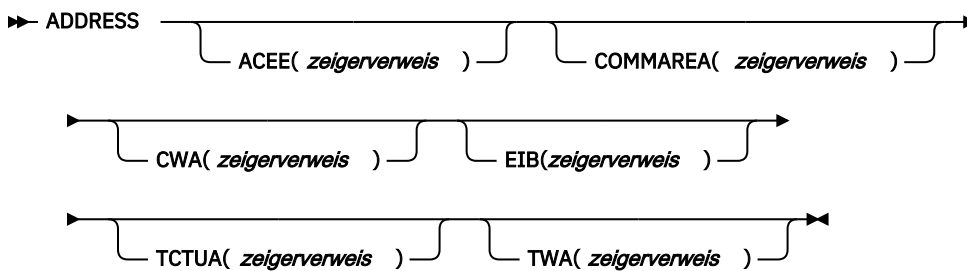
Das in der Option SUBEVENT angegebene Unterereignis ist ungültig. Dieser Fehler wird verursacht, wenn eines der folgenden Elemente als Unterereignis angegeben wird:

- Zusammengesetztes Ereignis
- Systemereignis
- Unterereignis eines anderen zusammengesetzten Ereignisses
- Unterereignis dieses zusammengesetzten Ereignisses, also ein bereits zu diesem zusammengesetzten Ereignis hinzugefügtes atomares Ereignis
- Eingabeereignis, falls das zusammengesetzte Ereignis den booleschen Operator AND verwendet

ADDRESS

Fordert Zugriff auf CICS-Speicherbereiche an.

ADDRESS



Dieser Befehl ist threadsicher.

Hinweis für dynamisches Transaktionsrouting: Die Verwendung von ADDRESS mit der Option CWA kann möglicherweise transaktionsübergreifende Affinitäten erstellen, die sich negativ auf die Verwendung des dynamischen Transaktionsroutings auswirken. Weitere Informationen zu Transaktionsaffinitäten finden Sie unter [Affinität](#).

Beschreibung

Mit ADDRESS kann auf die folgenden Bereiche zugegriffen werden:

- Das Zugriffssteuerungsumgebungselement (ACEE - Access Control Environment Element)
- Den Kommunikationsbereich, der für das aufgerufene Programm verfügbar ist (COMMAREA)
- Den gemeinsamen Arbeitsbereich (CWA - Common Work Area)
- Den EXEC-Schnittstellenblock (EIB - EXEC Interface Block)
- Den Benutzerbereich der Terminalsteuertabelle (TCTUA - Terminal Control Table User Area)
- Den Transaktionsarbeitsbereich (TWA - Transaction Work Area)

In Assemblersprache können nicht mehr als vier Optionen in einem Befehl ADDRESS angegeben werden.

Optionen

ACEE(zeigerverweis)

Diese Option gibt einen Zeiger auf das Zugriffssteuerungsumgebungselement zurück, den Steuerblock, der durch einen externen Sicherheitsmanager (ESM) generiert wird, wenn sich der Benutzer anmeldet. Wenn der Benutzer nicht angemeldet ist, wird die Adresse des ACEE für den CICS-Standardbenutzer (DFLTUSER) zurückgegeben. Wenn kein ACEE vorhanden ist, setzt CICS den Zeigerverweis auf den Nullwert X'FF000000'.

Informationen über die Zuordnung des ACEE-Datenbereichs finden Sie unter dem Zuordnungsmakro IHAACEE, das in SYS1.MACLIB bereitgestellt wird.

Anmerkung: Gehen Sie beim Zugriff auf einen ACEE in einem Serverprogramm, das durch einen Distributed Program Link (DPL) aufgerufen wird, mit Vorsicht vor. Die ACEE-Adresse, die zurückgegeben wird, hängt von der Verbindungssicherheit ab und kann nicht mit der Adresse des Benutzers, der am lokalen System angemeldet ist, identisch sein.

COMMAREA(zeigerverweis)

Diese Option gibt einen Zeigerverweis zurück, der auf die Adresse des Kommunikationsbereichs (COMMAREA) verweist, der für das gerade ausgeführte Programm verfügbar ist. Die Option COMMAREA dient zur Übergabe von Informationen zwischen Anwendungsprogrammen. Wenn der Kommunikationsbereich nicht vorhanden ist, wird der Zeigerverweis auf den Nullwert X'FF000000' gesetzt.

In C müssen Sie den Befehl ADDRESS COMMAREA verwenden, um die Adresse des Kommunikationsbereichs abzurufen, weil diese Adresse nicht als Argument an eine C-Hauptfunktion (Main) übergeben wird.

CWA(zeigerverweis)

Diese Option gibt einen Zeigerverweis zurück, der auf die Adresse des gemeinsamen Arbeitsbereichs (CWA) gesetzt ist. Dieser Bereich macht Informationen für Anwendungen verfügbar, die in einem einzelnen CICS-System ausgeführt werden. Wenn kein gemeinsamer Arbeitsbereich vorhanden ist, setzt CICS den Zeigerverweis auf den Nullwert X'FF000000'.

EIB(zeigerverweis)

Diese Option gibt einen Zeigerverweis zurück, der auf die Adresse des EXEC-Schnittstellenblocks (EIB) gesetzt ist. Sie müssen diese Option verwenden, um Adressierbarkeit für den EIB in anderen Anwendungsroutinen als der zuerst von CICS aufgerufenen Routine (in der die Adressierbarkeit für den EIB automatisch bereitgestellt wird) zu erreichen. Wenn das Anwendungsprogramm mit SYSEIB in der XOPTS-Parameterliste umgesetzt wird, gibt diese Option die Adresse des System-EIB zurück.

Wenn TASKDATALOC(ANY) in der Transaktionsdefinition definiert ist, kann die Adresse der Daten ober- oder unterhalb der 16-MB-Grenze liegen.

Wenn TASKDATALOC(BELOW) in der Transaktionsdefinition definiert ist und sich die Daten oberhalb der 16-MB-Grenze befinden, werden die Daten unter die 16-MB-Grenze kopiert und die Adresse dieser Kopie wird zurückgegeben.

C-Funktionen müssen Sie den Befehl ADDRESS EIB verwenden, um die Adresse des EXEC-Schnittstellenblocks abzurufen, weil diese Adresse nicht als Argument an eine C-Hauptfunktion (Main) übergeben wird. Sie müssen eine Anweisung ADDRESS EIB am Anfang jeder Anwendung codieren, wenn Sie auf den EIB zugreifen wollen oder wenn Sie einen Befehl verwenden, der die Option RESP oder RESP2 einschließt.

TCTUA(zeigerverweis)

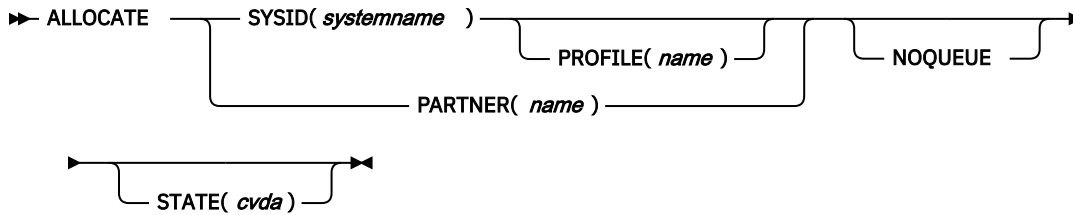
Diese Option gibt einen Zeigerverweis zurück, der auf die Adresse des Benutzerbereichs der Terminalsteuertabelle (TCTUA) für die Hauptfunktion und nicht für irgendeine andere Funktion gesetzt wird, die möglicherweise zugeordnet ist. Dieser Bereich wird zur Übergabe von Informationen zwischen Anwendungsprogrammen verwendet, jedoch nur, wenn den beteiligten Anwendungsprogrammen dasselbe Terminal zugeordnet ist. Wenn kein TCTUA vorhanden ist, wird der Zeigerverweis auf den Nullwert X'FF000000' gesetzt.

TWA(zeigerverweis)

Diese Option gibt einen Zeigerverweis zurück, der auf die Adresse des Transaktionsarbeitsbereichs (TWA) gesetzt ist. Dieser Bereich wird zur Übergabe von Informationen zwischen Anwendungspro-

Wenn TASKDATALOC(BELOW) in der Transaktionsdefinition definiert ist und sich die Daten oberhalb der 16-MB-Grenze befinden, werden die Daten unter die 16-MB-Grenze kopiert und die Adresse dieser Kopie wird zurückgegeben.

ALLOCATE (APPC)



Bedingungen: CBIDERR, INVREQ, NETNAMEIDERR, PARTNERIDERR, SYSBUSY, SYSIDERR

Beschreibung

Der Befehl ALLOCATE macht eine der Sitzungen, die dem genannten System zugeordnet sind, für das Anwendungsprogramm verfügbar und wählt optional eine Gruppe von Sitzungsverarbeitungsoptionen aus.

CICS gibt in EIBRSRCE im EIB die aus vier Byte bestehende Dialog-ID (CONVID) zurück, die das Anwendungsprogramm in allen nachfolgenden Befehlen, die sich auf den Dialog beziehen, verwendet.

Wenn keine Sitzung für die angeforderte logische APPC-Einheit (LU) verfügbar ist, wird die Anwendung ausgesetzt, bis eine Sitzung verfügbar wird. In diesem Fall kann die Aussetzung der Anwendung durch die Angabe der Option NOQUEUE oder NOSUSPEND vermieden werden. Das Schlüsselwort NOSUSPEND wird weiterhin als äquivalent zu NOQUEUE unterstützt, jedoch ist NOQUEUE das bevorzugte Schlüsselwort.

Eine Sitzung ist nur dann sofort für die Zuordnung verfügbar, wenn sie alle folgenden Bedingungen erfüllt:

- Konfliktgewinnersitzung
- Bereits gebunden
- Noch nicht zugeordnet

CICS versucht, eine Anforderung für eine Sitzung durch eine Sitzungsauswahl in der folgenden bevorzugten Reihenfolge zu erfüllen:

1. Konfliktgewinnersitzung, die gebunden, aber noch nicht zugeordnet ist (CICS ordnet sie zu). Dies ist eine Sitzung, die sofort verfügbar ist.
2. Konfliktgewinnersitzung, die nicht gebunden ist (CICS bindet sie und ordnet sie zu).
3. Konfliktverlierersitzung, die gebunden, aber noch nicht zugeordnet ist (CICS fordert das Nutzungsrecht für sie an).
4. Konfliktverlierersitzung, die nicht gebunden ist (CICS bindet sie und fordert das Nutzungsrecht für sie an).

Die Aktion, die von CICS ausgeführt wird, wenn keine Sitzung sofort verfügbar ist, hängt davon ab, ob Sie die Option NOQUEUE (oder die äquivalente Option NOSUSPEND) angeben, sowie davon, ob Ihre Anwendung einen Befehl HANDLE für die Bedingung SYSBUSY ausführt. In diesen Situationen führt CICS keine Anforderung des Nutzungsrechts (Bid) aus und bindet keine zusätzlichen Sitzungen. CICS sucht nach einer Sitzung, die sofort verfügbar ist (das heißt, eine Konfliktgewinnersitzung, die gebunden und noch nicht zugeordnet ist), und wenn keine verfügbar ist, wird die Bedingung SYSBUSY zurückgegeben. Die folgenden Kombinationen sind möglich:

Befehl HANDLE für Bedingung SYSBUSY abgesetzt

Der Befehl wird nicht in die Warteschlange gestellt und die Steuerung wird sofort an die Bezeichnung (Label) zurückgegeben, die im Befehl HANDLE angegeben wird, unabhängig davon, ob NOQUEUE angegeben wurde oder nicht.

Kein Befehl HANDLE für Bedingung SYSBUSY abgesetzt

Wenn Sie NOQUEUE (oder NOSUSPEND) angeben, wird die Anforderung nicht in die Warteschlange gestellt und die Steuerung wird sofort an Ihr Anwendungsprogramm zurückgegeben. Der SYSBUSY-Code (X'D3') wird im Feld EIBRCODE des EXEC-Schnittstellenblocks (EIB) angegeben. Sie sollten dieses Feld sofort nach Absetzen des Befehls ALLOCATE testen.

Daher hat der Befehl **HANDLE** für die Bedingung **SYSBUSY** den gleichen Effekt wie die Option **NOQUEUE**, abgesehen davon, an welche Position in der Anwendung die Steuerung zurückgegeben wird. Wenn der Befehl **HANDLE** verwendet wird, wird die Steuerung an die Bezeichnung (Label) zurückgegeben. Wird der Befehl nicht verwendet, wird die Steuerung an die Anweisung zurückgegeben, die auf den Befehl **ALLOCATE** folgt.

Wenn Sie die Option **NOQUEUE** nicht angeben und keinen Befehl **HANDLE** für die Option **SYSBUSY** absetzen, dann stellt CICS die Anforderung, wenn nicht sofort eine Sitzung verfügbar ist, in die Warteschlange (und Ihre Anwendung wartet), bis eine Sitzung verfügbar wird. Die Anforderung wird einer Sitzung zugeordnet, wenn entweder eine Gewinnersitzung verfügbar wird oder wenn CICS erfolgreich eine Verlierersitzung anfordert. Lassen Sie die Option **NOQUEUE** weg, wenn alle Gewinner- und Verlierersitzungen für die Zuordnung zu der Anforderung in Betracht gezogen werden sollen. Sie können die Attribute **QUEUELIMIT** und **MAXQTIME** der **CONNECTION**-Ressourcendefinitionen verwenden, um die Länge der Warteschlange von Anforderungen und die Zeit, die Anforderungen in der Warteschlange verbleiben, zu begrenzen. Unter [Warteschlangenmanagement bei systemübergreifenden Sitzungen](#) finden Sie weitere Informationen zu Zuordnungswarteschlangen. Der Wert **DTIMOUT**, der in der Transaktionsdefinition angegeben wird, kann dazu verwendet werden, die Wartezeit für einzelne Anforderungen zu begrenzen.

Optionen

NOQUEUE

Diese Option überschreibt die Standardaktion, wenn eine Bedingung **SYSBUSY** auftritt. Dies weist darauf hin, dass eine Sitzung nicht sofort verfügbar ist. Die Standardaktion besteht darin, die Anwendungsausführung auszusetzen, bis eine Sitzung verfügbar ist. Die Option **NOQUEUE** unterbindet dieses Warten. Die Steuerung kehrt sofort zu der Anweisung im Anwendungsprogramm zurück, die auf den Befehl folgt.

Wenn jedoch ein Befehl **HANDLE CONDITION** für **SYSBUSY** aktiv ist, wenn der Befehl ausgeführt wird, überschreibt dies die Standardaktion und die Steuerung wird an die Benutzerbezeichnung übergeben, die in **HANDLE CONDITION** angegeben wurde. Dieses Verhalten hat Vorrang vor der Option **NOQUEUE**, wird jedoch durch **NOHANDLE** oder **RESP** zurückgewiesen.

Wenn eine **APPC**-Anforderung **ALLOCATE** für eine einzelne Sitzungsverbindung von dem Konfliktverlierende abgesetzt wird, verursacht die Option **NOQUEUE** immer die Rückgabe einer Bedingung **SYSBUSY**, anstatt zuzulassen, dass die Anforderung versucht, die Sitzung zu erhalten. Wenn die Option **NOQUEUE** nicht angegeben wird, kann die Anforderung versuchen, die Sitzung zu erhalten.

Wenn eine **APPC**-Anforderung **ALLOCATE** für eine parallele Sitzungsverbindung abgesetzt wird und die Option **NOQUEUE** angegeben wird, können nur Sitzungen, die sofort verfügbar sind (d. h. ein Konfliktgewinner, der gebunden und noch nicht bereits zugeordnet ist) der Anforderung zugeordnet werden. Wenn keine solche Sitzung verfügbar ist, wird die Bedingung **SYSBUSY** zurückgegeben. Wenn die Option **NOQUEUE** nicht angegeben wird, kann die Anforderung versuchen, eine Verlierersitzung zu erhalten oder ungebundene Gewinnersitzungen zu binden.

PARTNER(name)

Diese Option gibt den Namen (8 Zeichen) einer Gruppe von Definitionen an, die die Namen einer fernen logischen Einheit (**NETNAME**) und eines Kommunikationsprofils enthalten, die in einer zugeordneten Sitzung verwendet werden sollen. Sie können diese Option als Alternative zur expliziten Angabe der Optionen **SYSID** und **PROFILE** verwenden.

PROFILE(name)

Diese Option gibt den Namen (1 - 8 Zeichen) einer Gruppe von Sitzungsverarbeitenden Optionen an, die während der Ausführung von **MAPPED**-Befehlen für die Sitzung verwendet werden sollen, die in der Option **SYSID** angegeben wird. Wenn Sie die Option **SYSID** angeben und die Option **PROFILE** nicht angeben, wird das Standardprofil (**DFHCICSA**) ausgewählt.

STATE(cvda)

Diese Option ruft den Status des aktuellen Dialogs ab. Der **CVDA**-Wert, der von CICS zurückgegeben wird, ist **ALLOCATED**.

SYSID(systemname)

Diese Option gibt den Namen (1 - 4 Zeichen) an, unter dem die ferne logische APPC-Einheit (LU) in diesem CICS bekannt ist. Diese Option fordert an, dass eine der Sitzungen zu dem benannten System zugeordnet werden soll.

Bedingungen**62 CBIDERR**

Diese Bedingung tritt auf, wenn das angeforderte Profil (PROFILE) nicht gefunden werden kann.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

Diese Bedingung tritt auf, wenn der Befehl ALLOCATE für die Einheit, an die er gerichtet wird, nicht gültig ist.

Standardaktion: Die Task wird abnormal beendet.

99 NETNAMEIDERR

Diese Bedingung tritt auf, wenn der im Parameter NETNAME angegebene Name der RDO-Definition für den im Befehl ALLOCATE angegebenen PARTNER ungültig ist.

Standardaktion: Die Task wird abnormal beendet.

97 PARTNERIDERR

Diese Bedingung tritt auf, wenn der Name, der in der Option PARTNER angegeben wurde, von CICS nicht erkannt wird.

Standardaktion: Die Task wird abnormal beendet.

59 SYSBUSY

Diese Bedingung tritt aus einer der folgenden Ursachen auf:

- Die Anforderung für eine Sitzung kann nicht sofort bedient werden. Dies ist nur dann möglich, wenn die Option NOQUEUE angegeben wurde oder ein Befehl HANDLE CONDITION für die Bedingung SYSBUSY aktiv ist.
- Der Befehl ALLOCATE wurde abgesetzt, als eine Wiederherstellung einer persistenten Sitzung noch in Verarbeitung war und die Sitzungen, die benötigt werden, um den Befehl zu erfüllen, noch nicht wiederhergestellt waren.

Standardaktion: Die Bedingung wird ignoriert.

53 SYSIDERR

Diese Bedingung tritt auf, wenn CICS das Anwendungsprogramm aus einer der folgenden Ursachen nicht mit einer passenden Sitzung bedienen kann:

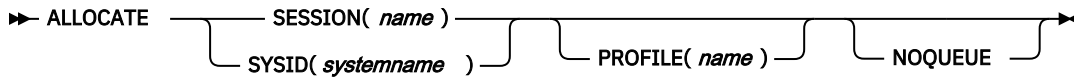
- Der in der Option SYSID angegebene Name wird von CICS nicht erkannt.
- Der Modusname, der aus der Option PROFILE abgeleitet wurde, ist keiner der Modusnamen, die für den APPC-Systemeintrag definiert sind.
- Alle Sitzungen in der Gruppe, die durch SYSID und den Modusnamen angegeben wird, sind außer Betrieb oder alle Sitzungen sind außer Betrieb.
- Der AID (automatischer Initiierungsdeskriptor), der Ihren Befehl ALLOCATE darstellt, wurde abgebrochen.
- Die Sitzungen sind belegt und die Zuordnungsanforderungen (in der Warteschlange) wurden bereinigt oder zurückgewiesen.

Standardaktion: Die Task wird abnormal beendet.

ALLOCATE (LUTYPE6.1)

Fordert eine Sitzung zu einer fernen logischen Einheit vom LU-Typ 6.1 (LUTYPE6.1) an.

ALLOCATE (LUTYPE6.1)



Bedingungen: CBIDERR, EOC, INVREQ, SESSBUSY, SESSIONERR, SYSBUSY, SYSIDERR

Beschreibung

Der Befehl ALLOCATE fordert eine alternative Einrichtung an und wählt optional eine Gruppe von Sitzungsverarbeitungsoptionen aus. Wenn die System-ID (SYSID) angegeben wird, macht CICS eine der Sitzungen für das Anwendungsprogramm verfügbar, die dem genannten System zugeordnet sind. Der Name dieser Sitzung kann aus dem Feld EIBRSRCE im EIB abgerufen werden. Wenn die Sitzung (SESSION) angegeben wird, macht CICS die genannte Sitzung verfügbar.

Wenn die angeforderte Sitzung nicht verfügbar ist, wird die Anwendung ausgesetzt, bis die Sitzung verfügbar wird. In diesem Fall kann die Aussetzung der Anwendung durch die Angabe der Option NOQUEUE oder NOSUSPEND vermieden werden. Das Schlüsselwort NOSUSPEND wird weiterhin als äquivalent zu NOQUEUE unterstützt, jedoch ist NOQUEUE das bevorzugte Schlüsselwort.

Optionen

NOQUEUE

Diese Option überschreibt die Standardaktion, wenn eine Bedingung SESSBUSY oder SYSBUSY auftritt. Diese Bedingungen geben an, dass die angeforderte Sitzung nicht sofort verfügbar ist. Die Standardaktion besteht darin, die Programmausführung auszusetzen, bis die Sitzung verfügbar ist. Die Option NOQUEUE unterbindet dieses Warten. Die Steuerung kehrt sofort zu der Anweisung im Anwendungsprogramm zurück, die auf den Befehl folgt.

Wenn jedoch ein Befehl HANDLE CONDITION für SESSBUSY oder SYSBUSY aktiv ist, wenn der Befehl ausgeführt wird, überschreibt dies die Standardaktion und die Steuerung wird an die Benutzerbezeichnung übergeben, die in HANDLE CONDITION angegeben wurde. Dieses Verhalten hat Vorrang vor der Option NOQUEUE, wird jedoch durch NOHANDLE oder RESP zurückgewiesen.

PROFILE(name)

Diese Option gibt den Namen (1-8 Zeichen) einer Gruppe von Sitzungsverarbeitenden Optionen an, die während der Ausführung von Terminalsteuerungsbefehlen für die Sitzung verwendet werden sollen, die in den Optionen SYSID oder SESSION angegeben wird. Wenn die Option PROFILE nicht angegeben wird, wird ein Standardprofil (DFHCICSA) ausgewählt.

SESSION(name)

Diese Option gibt die symbolische ID (1-4 Zeichen) für einen Terminaleintrag in der Terminalsteuertabelle (TCTTE) für die Sitzung an. Diese Option gibt die alternative Funktion an, die verwendet werden soll.

SYSID(systemname)

Diese Option gibt den Namen (1-4 Zeichen) eines TCTSE an. Diese Option gibt an, dass eine der Sitzungen zu dem benannten System zugeordnet werden soll.

Bedingungen

62 CBIDERR

Diese Bedingung tritt auf, wenn das angeforderte Profil (PROFILE) nicht gefunden werden kann.

Standardaktion: Die Task wird abnormal beendet.

06 EOC

Diese Bedingung tritt auf, wenn eine Anforderung/Antwort-Einheit (RU) empfangen wird, in der der Anzeiger für Kettenende (EOC - End-of-Chain) gesetzt ist. Das Feld EIBEOC enthält diesen Anzeiger ebenfalls.

Standardaktion: Die Bedingung wird ignoriert.

16 INVREQ

Diese Bedingung tritt auf, wenn die angegebene Sitzung dieser Task bereits zugeordnet ist oder wenn die Sitzung eine APPC-Sitzung ist.

Standardaktion: Die Task wird abnormal beendet.

60 SESSBUSY

Diese Bedingung tritt auf, wenn die Anforderung für die angegebene Sitzung nicht sofort bedient werden kann. Dies ist nur dann möglich, wenn die Option NOQUEUE angegeben wurde oder ein Befehl HANDLE CONDITION für die Bedingung SESSBUSY aktiv ist.

Standardaktion: Die Bedingung wird ignoriert.

58 SESSIONERR

Diese Bedingung tritt auf, wenn der in der Option SESSION angegebene Name kein Name eines TCTTE für eine LUTYPE6.1-Sitzung ist oder wenn die Sitzung nicht zugeordnet werden kann, weil sie außer Betrieb ist.

Standardaktion: Die Task wird abnormal beendet.

59 SYSBUSY

Diese Bedingung tritt aus einer der folgenden Ursachen auf:

- Die Anforderung für eine Sitzung kann nicht sofort bedient werden. Dies ist nur dann möglich, wenn die Option NOQUEUE angegeben wurde oder ein Befehl HANDLE CONDITION für die Bedingung SYSBUSY aktiv ist.
- Der Befehl ALLOCATE wurde abgesetzt, als eine Wiederherstellung einer persistenten Sitzung noch in Verarbeitung war und die Sitzungen, die benötigt werden, um den Befehl zu erfüllen, noch nicht wiederhergestellt waren.

Standardaktion: Die Bedingung wird ignoriert.

53 SYSIDERR

Diese Bedingung tritt auf, wenn CICS das Anwendungsprogramm aus einer der folgenden Ursachen nicht mit einer passenden Sitzung bedienen kann:

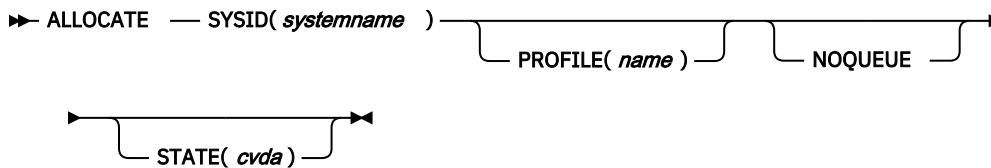
- Der in der Option SYSID angegebene Name wird von CICS nicht erkannt.
- Alle Sitzungen sind außer Betrieb.
- Der AID (automatischer Initiierungsdeskriptor), der Ihren Befehl ALLOCATE darstellt, wurde abgebrochen.
- Die Sitzungen sind belegt und die Zuordnungsanforderungen (in der Warteschlange) wurden bereinigt oder zurückgewiesen.

Standardaktion: Die Task wird abnormal beendet.

ALLOCATE (MRO)

Fordert eine MRO-Sitzung an.

ALLOCATE (MRO)



Bedingungen: INVREQ, SYSBUSY, SYSIDERR

Beschreibung

Der Befehl ALLOCATE fordert eine alternative Einrichtung an. CICS macht eine der Sitzungen, die dem in der Option SYSID benannten System zugeordnet sind, für das Anwendungsprogramm verfügbar. Der Name dieser Sitzung kann aus dem Feld EIBSRCE im EIB abgerufen werden.

Wenn die angeforderte Sitzung nicht verfügbar ist, wird die Anwendung ausgesetzt, bis die Sitzung verfügbar wird. In diesem Fall kann die Aussetzung der Anwendung durch die Angabe der Option NOQUEUE vermieden werden.

Optionen

NOQUEUE

Diese Option überschreibt die Standardaktion, wenn eine Bedingung SYSBUSY auftritt. Diese Bedingung gibt an, dass die angeforderte Sitzung nicht sofort verfügbar ist. Die Standardaktion besteht darin, die Anwendungsausführung auszusetzen, bis die Sitzung verfügbar ist. Die Option NOQUEUE unterbindet dieses Warten. Die Steuerung kehrt sofort zu der Anweisung im Anwendungsprogramm zurück, die auf den Befehl folgt.

Wenn jedoch ein Befehl HANDLE CONDITION für SYSBUSY aktiv ist, wenn der Befehl ausgeführt wird, überschreibt dies die Standardaktion und die Steuerung wird an die Benutzerbezeichnung übergeben, die in HANDLE CONDITION angegeben wurde. Dieses Verhalten hat Vorrang vor der Option NOQUEUE, wird jedoch durch NOHANDLE oder RESP zurückgewiesen.

PROFILE(name)

Diese Option gibt den Namen (1-8 Zeichen) einer Gruppe von Sitzungsverarbeitenden Optionen an, die während der Ausführung von Terminalsteuerungsbefehlen für die Sitzung verwendet werden sollen, die in der Option SYSID angegeben wird. Wenn die Option PROFILE nicht angegeben wird, wird ein Standardprofil (DFHCICSA) ausgewählt.

STATE(cvda)

Diese Option ruft den Status des aktuellen Dialogs ab. Der CVDA-Wert, der von CICS zurückgegeben wird, ist ALLOCATED.

SYSID(systemname)

Diese Option gibt den Namen (1-4 Zeichen) eines TCTSE an. Diese Option gibt an, dass eine der Sitzungen zu dem benannten System zugeordnet werden soll.

Bedingungen

16 INVREQ

Diese Bedingung tritt auf, wenn ein falscher Befehl für die verwendete logische Einheit (LU) oder das verwendete Terminal abgesetzt wurde.

Standardaktion: Die Task wird abnormal beendet.

59 SYSBUSY

Diese Bedingung tritt auf, wenn die Anforderung für eine Sitzung nicht sofort bedient werden kann. Dies ist nur dann möglich, wenn die Option NOQUEUE angegeben wurde oder ein Befehl HANDLE CONDITION für die Bedingung SYSBUSY aktiv ist.

Standardaktion: Die Bedingung wird ignoriert.

53 SYSIDERR

Diese Bedingung tritt auf, wenn CICS das Anwendungsprogramm aus einer der folgenden Ursachen nicht mit einer passenden Sitzung bedienen kann:

- Der in der Option SYSID angegebene Name wird von CICS nicht erkannt.
- Alle Sitzungen sind außer Betrieb.
- Der AID (automatischer Initiierungsdeskriptor), der Ihren Befehl ALLOCATE darstellt, wurde abgebrochen.
- Die Sitzungen sind belegt und die Zuordnungsanforderungen (in der Warteschlange) wurden bereitgestellt oder zurückgewiesen.

Standardaktion: Die Task wird abnormal beendet.

ASKTIME

Fordert ein aktuelles Datum und eine aktuelle Uhrzeit an.

ASKTIME



Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl ASKTIME aktualisiert das Datumsfeld (EIBDATE) und das CICS-Tageszeitfeld (EIBTIME) im EIB. Diese beiden Felder enthalten zu Anfang das Datum und die Uhrzeit des Zeitpunkts, zu dem die Task gestartet wurde.

Als Antwort auf einen Befehl ASKTIME setzt CICS ein MVS-STCK-Makro ab und ändert dieses durch den lokalen Zeitunterschied. Wenn zum Beispiel Ihre MVS-TOD-Systemuhr (Hardware) auf GMT-Zeit eingestellt ist und die Ortszeit als BST-Zeit (British Summer Time) definiert ist, wird die BST-Zeit im Feld EIBTIME gespeichert.

Weitere Informationen zum EIB finden Sie unter [EIB-Felder](#).

Optionen

ABSTIME(datenbereich)

Diese Option gibt den Datenbereich für die Anzahl Millisekunden seit dem 1. Januar 1900 (00:00 Uhr) an, der als absolute Zeit bezeichnet wird. Die Zeit wird der Uhrzeit der Systemuhr mit Anpassung im Hinblick auf Schaltsekunden und Ortszeitunterschiede (einschließlich Sommerzeit) entnommen, auf die Millisekunde abgeschnitten und als gepacktes Dezimalformat mit einer Länge von 8 Byte zurückgegeben.

Mit dem Befehl FORMATTIME können Sie die Daten in andere gängige Formate ändern.

Beispiel

Es wird zum Beispiel der folgende Befehl ausgeführt:

```
EXEC CICS ASKTIME ABSTIME(utime)
```

Anschließend enthält *utime* einen Wert, dessen Format ähnlich dem folgenden ist: 002837962864820.

Der Wert *datenbereich* hat folgendes Format:

```
COBOL: PIC S9(15) COMP-3  
C:      char data_area[8];  
PL/I:   FIXED DEC(15)  
ASM:    PL8
```

ASSIGN

Fordert Werte von außerhalb der lokalen Umgebung des Anwendungsprogramms an.

[illegible]

Bedingung: INVREQ
Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl **ASSIGN** ruft Werte von außerhalb der lokalen Umgebung des Anwendungsprogramms ab. Die Daten, die abgerufen werden, hängen von den angegebenen Optionen ab. Sie können bis zu 16 Optionen in einem Befehl **ASSIGN** angeben. Wenn Sie mehrere Optionen im Befehl **ASSIGN** angeben und irgendeine der angegebenen Optionen mit der Bedingung INVREQ fehlschlägt, werden die anderen angegebenen Optionen trotzdem mit den angeforderten Informationen gefüllt.

Für Optionen, die sich auf Terminals oder auf Daten von Terminals beziehen, gilt der Verweis immer für die Hauptfunktion.

Wenn die Hauptfunktion ein fernes Terminal ist, werden die Daten, die zurückgegeben werden, aus der lokalen Kopie der Informationen abgerufen. Die Anforderung wird nicht an das System geleitet, an das das ferne Terminal angeschlossen ist.

Das Transaktionsrouting erfolgt für den Befehl **ASSIGN** so weit wie möglich transparent. Im Allgemeinen sind die Werte, die zurückgegeben werden, identisch, unabhängig davon, ob die Transaktion lokal oder fern erfolgt.

Weitere Informationen zu diesen Optionen finden Sie unter [Entwicklung in einer systemübergreifenden Umgebung](#).

Optionen

ABCODE(datenbereich)

Diese Option gibt einen aktuellen Abbruchcode aus vier Zeichen zurück. Abbruchcodes sind in [Transaktionsabbruchcodes](#) dokumentiert. Wenn ein Abbruch erfolgt, wird die Variable auf Leerzeichen gesetzt.

ABDUMP(datenbereich)

Diese Option gibt einen 1 Byte großen Wert zurück. X 'FF' gibt an, dass ein Befehl **EXEC CICS ABEND ABCODE** ohne die Option NODUMP abgesetzt wurde und dass ABCODE einen Abbruchcode enthält. X '00' gibt an, dass entweder kein Speicherauszug generiert wurde oder ABCODE Leerzeichen enthält.

ABOFFSET(datenbereich)

Diese Option gibt einen Vollwort-Binärwert für den Offset in Byte eines Abbruchs zurück, wenn der letzte Abbruch mit dem Code ASRA, ASRB oder ASRD aufgetreten ist. Wenn der Abbruch außerhalb des aktuellen Programms aufgetreten ist, wird der Wert X 'FFFFFFFF' zurückgegeben. Der Datenbereich wird auf binäre Nullen gesetzt, wenn kein Abbruch mit dem Code ASRA, ASRB oder ASRD bei der Ausführung der absetzenden Transaktion aufgetreten ist oder wenn der Abbruch ursprünglich in einem fernen DPL-Serverprogramm aufgetreten ist.

ABPROGRAM(datenbereich)

Diese Option gibt einen acht Zeichen langen Namen des Programms mit dem Fehler für den letzten Abbruch zurück.

Wenn der Abbruch ursprünglich in einem DPL-Serverprogramm aufgetreten ist, das in einem fernen System ausgeführt wird, gibt die Option ABPROGRAM den Namen des DPL-Serverprogramms zurück.

Dieses Feld wird auf binäre Nullen gesetzt, wenn es nicht möglich ist, das Programm mit dem Fehler zum Zeitpunkt des Abbruchs zu ermitteln.

Wenn der letzte Abbruch vom Typ APCT (infolge eines nicht erfolgreichen Versuchs, ein Programm, eine Maskengruppe oder eine Partitionsgruppe zu laden) ist, wird der Name aus dem Programm, der Maskengruppe oder Partitionsgruppe genommen, das bzw. die nicht geladen wurde.

ACTIVITY(datenbereich)

Wenn dieses Programm für eine CICS-Transaktion einer BTS-Aktivität (BTS - Business Transaction Services) ausgeführt wird, gibt diese Option den aus 16 Zeichen bestehenden Namen der Aktivität zurück.

Eine Beschreibung von BTS finden Sie in [Übersicht über BTS](#).

ACTIVITYID(datenbereich)

Wenn dieses Programm für eine BTS-Aktivität ausgeführt wird, gibt diese Option die aus 52 Zeichen bestehende und durch CICS zugewiesene ID der Aktivitätsinstanz zurück.

Wenn ein Programm, das außerhalb des aktuellen Prozesses ausgeführt wird, die Steuerung dieser Aktivitätsinstanz anfordert, muss es diese ID in einem Befehl **ACQUIRE ACTIVITYID** angeben.

Eine Beschreibung von BTS finden Sie in [Übersicht über BTS](#).

ALTSCRNHT(datenbereich)

Diese Option gibt die alternative Anzeighöhe als Halbwort-Binärvariable zurück, die für das Terminal definiert ist. Wenn die Task nicht von einem Terminal aus initialisiert wurde, tritt die Bedingung INVREQ auf.

ALTSCRNWD(datenbereich)

Diese Option gibt die alternative Anzeigenbreite als Halbwort-Binärvariable zurück, die für das Terminal definiert ist. Wenn die Task nicht von einem Terminal aus initialisiert wurde, tritt die Bedingung INVREQ auf.

APLKYBD(datenbereich)

Diese Option gibt einen 1-Byte-Anzeiger zurück, der angibt, ob die Terminaltastatur die APL-Tastaturfunktion (X'FF') besitzt oder nicht (X'00'). Wenn die Task nicht von einem Terminal aus initialisiert wurde, tritt die Bedingung INVREQ auf.

APLTEXT(datenbereich)

Diese Option gibt einen 1-Byte-Anzeiger zurück, der angibt, ob die Terminaltastatur die APL-Textfunktion (X'FF') besitzt oder nicht (X'00'). Wenn die Task nicht von einem Terminal aus initialisiert wurde, tritt die Bedingung INVREQ auf.

APPLICATION(datenbereich)

Diese Option gibt den aus 64 Zeichen bestehenden Namen der aktuellen Anwendung zurück, die der Task zugeordnet ist. Sie ist Teil des Anwendungskontexts, der aus dem Anwendungsnamen, dem Plattformnamen und dem Operationsnamen sowie aus der Haupt-, Neben- und Mikroversionsnummer der Anwendung besteht. Wenn der Task kein Anwendungskontext zugeordnet ist, werden Leerzeichen zurückgegeben.

APPLID(datenbereich)

Diese Option gibt eine acht Zeichen lange Anwendungs-ID (APPLID) des CICS-Systems zurück, das Eigner der Transaktion ist.

Wenn Ihr System XRF verwendet, ist der Wert, der zurückgegeben wird, eine generische Anwendungs-ID (APPLID). Ein Anwendungsprogramm bleibt von einer Übernahme des aktiven durch den alternativen Wert unberührt.

ASRAINTRPT(datenbereich)

Diese Option gibt einen Datenbereich aus acht Zeichen zurück, der den ILC-Code (Instruktionslängencode) und den PIC-Code (PIC - Programm Interrupt Code, Programmunterbrechungscode) zu dem Zeitpunkt enthält, als der letzte Abbruch mit dem Code AICA, ASRA, ASRB, ASRD oder ASRE aufgetreten ist. Das Feld enthält binäre Nullen, wenn kein Abbruch mit dem Code AICA, ASRA, ASRB, ASRD oder ASRE bei der Ausführung der absetzenden Transaktion aufgetreten ist oder wenn der Abbruch ursprünglich in einem fernen DPL-Serverprogramm aufgetreten ist. Wenn gültig, sieht der Inhalt der acht zurückgegebenen Byte wie folgt aus:

- ILC (2 Byte binär)
- PIC (2 Byte binär)
- Füller (4 Byte binär, immer null)

ASRAKEY(cvda)

Diese Option gibt den Ausführungsschlüssel zum Zeitpunkt des letzten Abbruchs mit AEYD, AEYF, AICA, ASRA oder ASRB zurück, sofern zutreffend. Die folgenden CVDA-Werte sind möglich:

CICSEXECKEY

Dieser Wert wird zurückgegeben, wenn die Task zum Zeitpunkt des letzten Abbruchs mit AEYD, AEYF, AICA, ASRA oder ASRB in einem CICS-Schlüssel ausgeführt wurde. Wenn der CICS-Subsystemspeicherschutz nicht aktiv ist, werden alle Programme in einem CICS-Schlüssel ausgeführt.

USEREXECKEY

Dieser Wert wird zurückgegeben, wenn die Task zum Zeitpunkt des letzten Abbruchs mit AEYD, AEYF, AICA, ASRA oder ASRB in einem Benutzerschlüssel ausgeführt wurde.

NONCICS

Dieser Wert wird zurückgegeben, wenn der Ausführungsschlüssel zum Zeitpunkt des letzten Abbruchs keiner der CICS-Schlüssel war, zum Beispiel nicht Schlüssel 8 oder Schlüssel 9.

NOTAPPLIC

Dieser Wert wird zurückgegeben, wenn kein Abbruch mit AEYD, AEYF, AICA, ASRA oder ASRB vorliegt.

ASRAPSW(datenbereich)

Diese Option gibt einen Datenbereich aus acht Zeichen zurück, der das Programmstatuswort (PSW) zu dem Zeitpunkt enthält, als der letzte Abbruch mit dem Code AICA, ASRA, ASRB, ASRD oder ASRE aufgetreten ist.

Das Feld enthält binäre Nullen, wenn kein Abbruch mit dem Code AICA, ASRA, ASRB, ASRD oder ASRE bei der Ausführung der absetzenden Transaktion aufgetreten ist oder wenn der Abbruch ursprünglich in einem fernen DPL-Serverprogramm aufgetreten ist.

ASRAPSW16(datenbereich)

Diese Option gibt einen Datenbereich aus 16 Zeichen zurück, der das 128-Bit-Programmstatuswort (PSW) zu dem Zeitpunkt enthält, als der letzte Abbruch mit dem Code AICA, ASRA, ASRB, ASRD oder ASRE aufgetreten ist.

Das Feld enthält binäre Nullen, wenn kein Abbruch mit dem Code AICA, ASRA, ASRB, ASRD oder ASRE bei der Ausführung der absetzenden Transaktion aufgetreten ist oder wenn der Abbruch ursprünglich in einem fernen DPL-Serverprogramm aufgetreten ist.

ASRAREGS(datenbereich)

Diese Option gibt den Inhalt der Allgemeinregister 0 - 15 zu dem Zeitpunkt zurück, als der letzte Abbruch mit dem Code AICA, ASRA, ASRB, ASRD oder ASRE aufgetreten ist.

Der Inhalt der Register wird im Datenbereich (64 Byte lang) in der Reihenfolge 0, 1, ..., 14, 15 zurückgegeben.

Der Datenbereich wird auf binäre Nullen gesetzt, wenn kein Abbruch mit dem Code AICA, ASRA, ASRB, ASRD oder ASRE bei der Ausführung der absetzenden Transaktion aufgetreten ist oder wenn der Abbruch ursprünglich in einem fernen DPL-Serverprogramm aufgetreten ist.

ASRAREGS64(datenbereich)

Diese Option gibt den Inhalt der 64-Bit-Allgemeinregister 0 - 15 zu dem Zeitpunkt zurück, als der letzte Abbruch mit dem Code AICA, ASRA, ASRB, ASRD oder ASRE aufgetreten ist.

Der Inhalt der Register wird im Datenbereich (128 Byte lang) in der Reihenfolge 0, 1, ..., 14, 15 zurückgegeben.

Der Datenbereich wird auf binäre Nullen gesetzt, wenn kein Abbruch mit dem Code AICA, ASRA, ASRB, ASRD oder ASRE bei der Ausführung der absetzenden Transaktion aufgetreten ist oder wenn der Abbruch ursprünglich in einem fernen DPL-Serverprogramm aufgetreten ist.

ASRASPC(cvda)

Gibt den Typ von Bereich zurück, der die Steuerung zum Zeitpunkt des letzten Abbruchs mit AEYD, AEYF, AICA, ASRA oder ASRB besitzt, sofern zutreffend. Die folgenden CVDA-Werte sind möglich:

SUBSPACE

Dieser Wert wird zurückgegeben, wenn die Task zum Zeitpunkt des letzten Abbruchs mit AEYD, AEYF, AICA, ASRA oder ASRB entweder im eigenen Unterbereich oder im gemeinsamen Unterbereich ausgeführt wurde.

BASESPACE

Dieser Wert wird zurückgegeben, wenn die Task zum Zeitpunkt des letzten Abbruchs mit AEYD, AEYF, AICA, ASRA oder ASRB im Basisbereich ausgeführt wurde. Alle Tasks werden im Basisbereich ausgeführt, wenn die Transaktionsisolation nicht aktiv ist.

NOTAPPLIC

Dieser Wert wird zurückgegeben, wenn kein Abbruch mit AEYD, AEYF, AICA, ASRA oder ASRB vorliegt.

ASRASTG(*cvda*)

Diese Option gibt den Typ von Speicher an, der zum Zeitpunkt des letzten Abbruchs mit AEYD, AEYF, AICA, ASRA oder ASRB adressiert wurde, sofern zutreffend. Die folgenden CVDA-Werte sind möglich:

CICS

Dieser Wert wird zurückgegeben, wenn der Speicher, der adressiert wird, CICS-Schlüsselspeicher ist. Dieser kann sich in einem der dynamischen CICS-Speicherbereiche (CDSA, ECDSA, ETDSA oder GCDSA) befinden. Dieser kann sich in einem der schreibgeschützten dynamischen Speicherbereiche (RDSA oder ERDSA) befinden, wenn CICS mit der Option NOPROTECT im Systeminitialisierungsparameter **RENTPGM** ausgeführt wird oder wenn der Speicherschutz nicht aktiv ist.

USER

Dieser Wert wird zurückgegeben, wenn der Speicher, der adressiert wird, Benutzerschlüsselspeicher in einem der dynamischen Benutzerspeicherbereiche (UDSA, EUDSA oder GUDSA) ist.

READONLY

Dieser Wert wird zurückgegeben, wenn der Speicher, der adressiert wird, schreibgeschützter Speicher in einem der schreibgeschützten dynamischen Speicherbereiche (RDSA oder ERDSA) ist, wenn CICS mit der Option PROTECT im Systeminitialisierungsparameter **RENTPGM** ausgeführt wird.

NOTAPPLIC

Dieser Wert wird unter den folgenden Bedingungen zurückgegeben:

- Es wurde kein Abbruch mit AEYD, AEYF, AICA, ASRA oder ASRB für diese Task gefunden.
- Der betroffene Speicher in einem Abbruch wird nicht von CICS verwaltet.
- Der Abbruch mit ASRA wurde nicht durch einen Abbruch OC4 verursacht.

BRIDGE(*datenbereich*)

Diese Option gibt die aus vier Zeichen bestehende Transaktions-ID (TRANSID) der Brückenüberwachungstransaktion zurück, die einen Befehl **START BREXIT TRANSID** abgesetzt hat, um die Benutzertransaktion zu starten, die diesen Befehl abgesetzt hat. In den folgenden Situationen werden Leerzeichen zurückgegeben:

- Die Benutzertransaktion wurde nicht durch eine Brückenüberwachungstransaktion gestartet.
- Dieser Befehl wurde durch ein Programm abgesetzt, das von einer Distributed Program Link-Anforderung (DPL-Anforderung) gestartet wurde.

Anmerkung: Wenn der Befehl **START BREXIT** über einen Brückenexit abgesetzt wurde, wird die Transaktions-ID (TRANSID) der Brückenüberwachung zurückgegeben, die einen Befehl **START BREXIT** unter Angabe des Brückenexits abgesetzt hat.

BTRANS(*datenbereich*)

Diese Option gibt einen 1-Byte-Anzeiger zurück, der angibt, ob das Terminal mit der Hintergrundtransparenzfunktion (X'FF') definiert ist oder nicht (X'00'). Wenn die Task nicht von einem Terminal aus initialisiert wurde, tritt die Bedingung INVREQ auf.

CHANNEL(*datenbereich*)

Diese Option gibt den aus 16 Zeichen bestehenden Namen des aktuellen Kanals des Programms, sofern vorhanden, zurück. Andernfalls gibt sie Leerzeichen zurück.

CMDSEC(*datenbereich*)

Diese Option gibt einen 1-Byte-Anzeiger zurück, der angibt, ob für die aktuelle Task die Prüfung der Befehlssicherheit definiert ist. (X für ja, leer für nein.)

COLOR(datenbereich)

Diese Option gibt einen 1-Byte-Anzeiger zurück, der angibt, ob das Terminal mit der erweiterten Farbfunktion (X'FF') definiert ist oder nicht (X'00'). Wenn die Task nicht von einem Terminal aus initialisiert wurde, tritt die Bedingung INVREQ auf.

CWALENG(datenbereich)

Diese Option gibt ein Halbwort-Binärfeld zurück, das die Länge des gemeinsamen Arbeitsbereichs (CWA) angibt. Wenn kein CWA vorhanden ist, wird die Länge null zurückgegeben.

DEFSCRNHT(datenbereich)

Diese Option gibt eine Halbwort-Binärvariable zurück, die die Standardanzeighöhe enthält, die für das Terminal definiert ist. Wenn die Task nicht von einem Terminal aus initialisiert wurde, tritt die Bedingung INVREQ auf.

DEFSCRNWD(datenbereich)

Diese Option gibt eine Halbwort-Binärvariable zurück, die die Standardanzeigenbreite enthält, die für das Terminal definiert ist. Wenn die Task nicht von einem Terminal aus initialisiert wurde, tritt die Bedingung INVREQ auf.

DELIMITER(datenbereich)

Diese Option gibt ein 1 Byte großes Datenübertragungssteuerzeichen (Data-Link) für eine 3600-Einheit zurück. Die folgenden Werte sind möglich:

X'80'

Eingabe wurde durch ein Textendezeichen (ETX) beendet.

X'40'

Eingabe wurde durch ein Blockendezeichen (ETB) beendet.

X'20'

Eingabe wurde durch ein Datensatztrennzeichen (IRS - Inter-Record Separator) beendet.

X'10'

Eingabe wurde durch ein Headerstartzeichen (SOH - Start of Header) beendet.

X'08'

Transparente Eingabe.

Wenn die Task nicht von einem Terminal aus initialisiert wurde, tritt die Bedingung INVREQ auf.

DESTCOUNT(datenbereich)

Diese Option gibt ein Halbwort-Binärfeld zurück. Diese Option hat die folgenden Zwecke:

- Nach einem BMS-Befehl **ROUTE** zeigt es an, dass der erforderliche Wert die Anzahl der verschiedenen Terminaltypen in der Routing-Liste ist und daher die Anzahl der Überlaufsteuerbereiche, die möglicherweise erforderlich sind.
- Innerhalb der BMS-Überlaufverarbeitung zeigt es an, dass der erforderliche Wert die relative Überlaufsteuer Nummer des Ziels ist, das einen Überlauf festgestellt hat. Wenn diese Option angegeben wird, wenn die Überlaufverarbeitung nicht aktiv ist, hat der Wert, der abgerufen wird, keine Bedeutung. Wenn keine BMS-Befehle abgesetzt werden, tritt die Bedingung INVREQ auf.

DESTID(datenbereich)

Diese Option gibt eine 8-Byte-ID des externen Ziels auf der rechten Seite mit Leerzeichen bis auf acht Zeichen aufgefüllt zurück. Wenn diese Option angegeben wird, bevor ein Stapeldatenaustauschbefehl in der Task abgesetzt wird, tritt die Bedingung INVREQ auf.

DESTIDLENG(datenbereich)

Diese Option gibt eine Halbwort-Binärzahl für die Länge der Ziel-ID, zurück, die durch DESTID abgerufen wird. Wenn diese Option angegeben wird, bevor ein Stapeldatenaustauschbefehl in der Task abgesetzt wird, tritt die Bedingung INVREQ auf.

DSSCS(datenbereich)

Diese Option gibt einen 1-Byte-Anzeiger zurück, der angibt, ob die Hauptfunktion eine SCS-Basisdatenstromeinheit (X'FF') ist oder nicht (X'00').

Wenn die Task nicht von einem Terminal aus initialisiert wurde, tritt die Bedingung INVREQ auf.

DS3270(datenbereich)

Diese Option gibt einen 1-Byte-Anzeiger zurück, der angibt, ob die Hauptfunktion eine 3270-Datenstromeinheit (X'FF') ist oder nicht (X'00').

Wenn die Task nicht von einem Terminal aus initialisiert wurde, tritt die Bedingung INVREQ auf.

ERRORMSG(datenbereich)

Diese Option gibt eine Fehlernachricht mit einer Länge von maximal 500 Byte zurück, auf die im Transaktionsabbruchsteuerblock für die CICS-Task verwiesen wird. Nach einem Fehler einer DPL-Anforderung ist dies die Nachricht, die vom fernen System zurückgegeben wird. Nachrichten, die kürzer als 500 Byte sind, werden mit Nullen aufgefüllt.

Wenn keine Nachricht vorhanden ist, enthält der 500-Byte-Bereich Nullen.

ERRORMSGLEN(datenbereich)

Diese Option gibt einen Halbwort-Binärwert zurück, der die Länge der Nachricht darstellt, die für ERRORMSG zurückgegeben wird. Wenn die Nachricht, auf die im Transaktionsabbruchsteuerblock verwiesen wird, die Länge von 500 Byte überschreitet, wird die Nachricht abgeschnitten und die Länge auf 500 gesetzt.

Wenn keine Nachricht vorhanden ist, wird die Länge 0 zurückgegeben.

EWASUPP(datenbereich)

Diese Option gibt einen 1-Byte-Anzeiger zurück, der angibt, ob die Erase-Write-Alternative (EWA) unterstützt wird (X'FF') oder nicht (X'00').

Wenn die Task nicht von einem Terminal aus initialisiert wurde, tritt die Bedingung INVREQ auf.

EXTDS(datenbereich)

Diese Option gibt einen 1-Byte-Anzeiger zurück, der angibt, ob das Terminal den erweiterten 3270-Datenstrom akzeptiert (X'FF') oder nicht (X'00'). Für ein Terminal, das die Abfragefunktion, Farbfunktion, erweiterte Hervorhebung, programmierte Symbole oder die Prüfung (Validierung) unterstützt, ist die erweiterte Datenstromfunktionalität erforderlich. Für ein Terminal, das den Befehl mit strukturiertem Abfragefeld akzeptiert, wird dieser Anzeiger ebenfalls gesetzt. Wenn der erweiterte Datenstrom aktiv ist, unterstützt die Einheit den Befehl WRITE mit strukturiertem Feld und das strukturierte Ausgangsabfragefeld.

Weitere Informationen zu strukturierten Abfragefeldern finden Sie in der Veröffentlichung [IBM 3270 Data Stream Programmers Reference](#).

Wenn die Task nicht von einem Terminal aus initialisiert wurde, tritt die Bedingung INVREQ auf.

FACILITY(datenbereich)

Diese Option gibt eine 4-Byte-ID der Hauptfunktion zurück, die die Transaktion initialisiert hat, die diesen Befehl absetzt. Wenn diese Option angegeben wird und keine Funktion zugeordnet ist, tritt eine Bedingung INVREQ auf.

Anmerkung: Sie können die Option QNAME verwenden, um den Namen der partitionsinternen Warteschlange für transiente Daten abzurufen, wenn die Transaktion durch den Ablauf einer Auslöserebene für transiente Daten initialisiert wurde.

FCI(datenbereich)

Diese Option gibt einen 1-Byte-Anzeiger für die Funktionssteuerung zurück. Weitere Informationen finden Sie unter [Von ASSIGN zurückgegebene Codes](#). Dies gibt den Typ von Funktion an, der der Transaktion zugeordnet ist. Beispiel: X'01' gibt ein Terminal oder eine logische Einheit an. Der abgerufene Wert wird immer zurückgegeben.

GCHARS(datenbereich)

Diese Option gibt einen Halbwort-Binärwert für die globale ID für Grafikzeichensatz (die GCSGID) zurück. Der Wert ist eine Zahl aus dem Bereich 1 - 65534, der einen Satz von Grafikzeichen darstellt, die auf dem Terminal ein- oder ausgegeben werden können. Wenn die Task nicht von einem Terminal aus initialisiert wurde, tritt die Bedingung INVREQ auf.

GCODES(datenbereich)

Diese Option gibt einen Halbwort-Binärwert für die globale Codepage-ID (die CPGID) zurück. Der Wert ist eine Zahl aus dem Bereich 1 - 65534, die die EBCDIC- oder ASCII-Codepage darstellt, die die Codpunkte für die Zeichen definiert, die auf dem Terminal ein- oder ausgegeben werden können. Wenn die Task nicht von einem Terminal aus initialisiert wurde, tritt die Bedingung INVREQ auf.

GMMI(datenbereich)

Diese Option gibt einen 1-Byte-Anzeiger zurück, der angibt, ob eine "Guten-Morgen-Nachricht" für das Terminal, das der aktiven Transaktion zugeordnet ist, gilt (X'FF') oder nicht (X'00'). Wenn diese Option angegeben wird und die aktuelle Task keinem Terminal zugeordnet ist, tritt die Bedingung INVREQ auf.

HILIGHT(datenbereich)

Diese Option gibt einen 1-Byte-Anzeiger zurück, der angibt, ob das Terminal mit der erweiterten Hervorhebungsfunktion definiert ist (X'FF') oder nicht (X'00'). Wenn die Task nicht von einem Terminal aus initialisiert wurde, tritt die Bedingung INVREQ auf.

INITPARM(datenbereich)

Diese Option gibt einen 60 Zeichen großen Datenbereich zurück, der alle Initialisierungsparameter enthält, die für das Programm im Initialisierungsparameter **INITPARM** angegeben wurden. Die Werte werden nur dann zurückgegeben, wenn der Name des Programms, das den Befehl absetzt, mit einem Programmnamen übereinstimmt, der im Systeminitialisierungsparameter **INITPARM** angegeben ist. Wenn für das Programm keine Parameter vorhanden sind, wird der Bereich nicht aktualisiert und sein Inhalt ist nicht definiert. Verwenden Sie **INITPARM** zusammen mit INITPARMLEN, um zu ermitteln, ob ein Parameter angegeben wurde oder nicht. Der Wert, der durch **ASSIGN INITPARM** allein zurückgegeben wird, kann nicht zur Bestimmung verwendet werden, ob ein Systeminitialisierungsparameter angegeben wurde.

INITPARMLEN(datenbereich)

Diese Option gibt einen Halbwort-Binärwert für die Länge von INITPARM zurück. Wenn keine Parameter für INITPARM vorhanden sind, enthält INITPARMLEN binäre Nullen.

INPARTN(datenbereich)

Diese Option gibt den ein oder zwei Zeichen langen Namen der letzten Eingabepartition zurück. Wenn noch keine Maske positioniert ist, wenn das BMS-Routing aktiv ist oder wenn die Task nicht über ein Terminal initialisiert wurde, tritt die Bedingung INVREQ auf.

INPUTMSGLEN(datenbereich)

Diese Option gibt einen Halbwort-Binärwert für die Länge der Terminaleingabezeichenfolge in Byte zurück. Wenn keine Terminaleingabedaten vorhanden sind, wird die Länge null zurückgegeben.

INVOKINGPROG(datenbereich)

Diese Option gibt den aus acht Zeichen bestehenden Namen des Anwendungsprogramms zurück, das den Befehl **LINK** oder **XCTL** verwendet hat, um eine Verbindung mit dem aktuellen Programm herzustellen bzw. die Steuerung an das aktuelle Programm zu übertragen:

- Wenn Sie den Befehl **ASSIGN INVOKINGPROG** in einem fernen Programm absetzen, das durch einen Distributed Program Link-Befehl (DPL-Befehl) aufgerufen wurde, gibt CICS den Namen des Programms zurück, das den DPL-Befehl abgesetzt hat.
- Wenn Sie den Befehl **ASSIGN INVOKINGPROG** in einem Anwendungsprogramm auf der höchsten Ebene absetzen, gibt CICS acht Leerzeichen zurück.
- Wenn Sie den Befehl **ASSIGN INVOKINGPROG** in einem durch den Benutzer austauschbaren Programm, einem Brückexitprogramm oder einem Programm der Programmlistentabelle absetzen, gibt CICS acht Leerzeichen zurück.
- Wenn Sie den Befehl **ASSIGN INVOKINGPROG** über einen globalen Benutzerexit, einen taskbezogenen Exit oder ein Anwendungsprogramm absetzen, das über einen solchen Exit verbunden wurde, gibt CICS den Namen des letzten aufrufenden Programms zurück, das kein globaler Benutzerexit oder taskbezogener Benutzerexit war.

KATAKANA(datenbereich)

Diese Option gibt einen 1-Byte-Anzeiger zurück, der angibt, ob die Hauptfunktion Katakana unterstützt (X'FF') oder nicht (X'00'). Wenn die Task nicht von einem Terminal aus initialisiert wurde, tritt die Bedingung INVREQ auf.

LANGINUSE(datenbereich)

Diese Option gibt einen drei Byte langen mnemonischen Code zurück, der die verwendete Sprache angibt. Der drei Byte lange mnemonische Code hat eine 1:1-Entsprechung zur ein Byte langen Option NATLANGINUSE. Mögliche Werte des Codes finden Sie unter [Codes für Landessprachen](#).

LDCMNEM(datenbereich)

Diese Option gibt einen zwei Byte langen mnemonischen logischen Einheitencode (LDC - Logical Device Code) für das Ziel zurück, das den Überlauf festgestellt hat. Wenn diese Option angegeben wird, wenn die Überlaufverarbeitung nicht aktiv ist, hat der Wert, der abgerufen wird, keine Bedeutung. Wenn keine BMS-Befehle abgesetzt werden, tritt die Bedingung INVREQ auf.

LDCNUM(datenbereich)

Diese Option gibt einen ein Byte langen numerischen LDC-Wert für das Ziel zurück, das den Überlauf festgestellt hat. Dieser Code gibt den Typ des logischen Einheitencodes an, wie zum Beispiel Drucker oder Konsole. Wenn diese Option angegeben wird, wenn die Überlaufverarbeitung nicht aktiv ist, hat der Wert, der abgerufen wird, keine Bedeutung.

LINKLEVEL(datenbereich)

Diese Option gibt einen Halbwort-Binärwert zurück, der die Programmverbindungsebene im lokalen System darstellt. Die höchste Verbindungsebene ist 1. Für jeden Befehl EXEC CICS LINK wird die Verbindungsebene um 1 erhöht. Die Verbindungsebene wird nicht für eine Sprachenanweisung CALL erhöht. Wenn ein Programm das Ziel einer DPL-Anforderung ist, wird die Verbindungsebene zurückgegeben, die innerhalb der CICS-Region gilt, die ausgeführt wird, und nicht die weiter verteilte Transaktion. Wenn auf ein Programm über DPL zugegriffen wird, ist die Verbindungsebene 1 das CICS-Spiegelprogramm DFHMIRS.

LOCALCCSID(datenbereich)

Diese Option gibt den Vollwort-Binärwert für die ID des codierten Zeichensatzes (Coded Character Set Identifier - CCSID) zurück, die für die CICS-Region verwendet wird. Dabei handelt es sich um den Wert im Systeminitialisierungsparameter **LOCALCCSID**.

MAJORVERSION(datenbereich)

Diese Option gibt den Vollwort-Binärwert zurück, der die Hauptversion der aktuellen Anwendung darstellt, die der Task zugeordnet ist und die Teil des Anwendungskontexts ist. Wenn der Task kein Anwendungskontext zugeordnet ist, wird der Wert -1 zurückgegeben.

MAPCOLUMN(datenbereich)

Diese Option gibt die Halbwort-Binärnummer für die Spalte auf der Anzeige zurück, die den Ursprung der zuletzt positionierten Maske enthält. Wenn noch keine Maske positioniert ist, wenn das BMS-Routing aktiv ist oder wenn die Task nicht über ein Terminal initialisiert wurde, tritt die Bedingung INVREQ auf.

MAPHEIGHT(datenbereich)

Diese Option gibt einen Halbwort-Binärwert für die Höhe der zuletzt positionierten Maske zurück. Wenn noch keine Maske positioniert ist, wenn das BMS-Routing aktiv ist oder wenn die Task nicht über ein Terminal initialisiert wurde, tritt die Bedingung INVREQ auf.

MAPLINE(datenbereich)

Diese Option gibt die Halbwort-Binärnummer für die Zeile auf der Anzeige zurück, die den Ursprung der zuletzt positionierten Maske enthält. Wenn noch keine Maske positioniert ist, wenn das BMS-Routing aktiv ist oder wenn die Task nicht über ein Terminal initialisiert wurde, tritt die Bedingung INVREQ auf.

MAPWIDTH(datenbereich)

Diese Option gibt einen Halbwort-Binärwert für die Breite der zuletzt positionierten Maske zurück. Wenn noch keine Maske positioniert ist, wenn das BMS-Routing aktiv ist oder wenn die Task nicht über ein Terminal initialisiert wurde, tritt die Bedingung INVREQ auf.

MICROVERSION(datenbereich)

Diese Option gibt den Vollwort-Binärwert zurück, der die Mikroversion der aktuellen Anwendung darstellt, die der Task zugeordnet ist und die Teil des Anwendungskontexts ist. Wenn der Task kein Anwendungskontext zugeordnet ist, wird der Wert -1 zurückgegeben.

MINORVERSION(datenbereich)

Diese Option gibt den Vollwort-Binärwert zurück, der die Nebenversion der aktuellen Anwendung darstellt, die der Task zugeordnet ist und die Teil des Anwendungskontexts ist. Wenn der Task kein Anwendungskontext zugeordnet ist, wird der Wert -1 zurückgegeben.

MSRCONTROL(datenbereich)

Diese Option gibt einen 1-Byte-Anzeiger zurück, der angibt, ob das Terminal die Steuerung eines Magnetcodelesers (MSR - Magnetic Slot Reader) unterstützt (X'FF') oder nicht (X'00'). Wenn die Task nicht von einem Terminal aus initialisiert wurde, tritt die Bedingung INVREQ auf.

NATLANGINUSE(datenbereich)

Diese Option gibt einen ein Byte langen mnemonischen Code zurück, der die Landessprache angibt, die der Benutzer-ID (USERID) für die aktuelle Task zugeordnet ist (was die Standard-Benutzer-ID sein könnte). Eine Erläuterung der Ableitung dieses Werts finden Sie in den Informationen zum Befehl **SIGNON**. (NATLANGINUSE zeigt nicht die Systemstandardsprache, wie sie im Systeminitialisierungsparameter **NATLANG** angegeben wurde.)

Mögliche Werte des Codes finden Sie unter [Codes für Landessprachen](#).

NETNAME(datenbereich)

Diese Option gibt den acht Zeichen langen Namen der logischen Einheit im z/OS Communications Server-Netz zurück. Wenn die Task nicht von einem Terminal aus initialisiert wurde, tritt die Bedingung INVREQ auf. Ist die Hauptfunktion kein logisches Terminal, gibt CICS keine Nullzeichenfolge mehr zurück, sondern den Netznamen des fernen Terminals.

Wenn dieser Befehl von einer Benutzertransaktion abgesetzt wurde, die durch eine 3270-Brückentransaktion gestartet wurde, ist der zurückgegebene Wert die Terminal-ID (TERMID) der Brückeneinrichtung.

Wenn die CICS-Region LU-Alias für z/OS Communications Server unterstützt, ist der Netzname (NETNAME), der von CICS zurückgegeben wird, möglicherweise ein LU-Alias, der entweder dynamisch von z/OS Communications Server zugeordnet wurde oder im Parameter **LUALIAS** einer CDRSC-Definition vordefiniert wurde.

NEXTTRANSID(datenbereich)

Diese Option gibt eine vier Zeichen lange ID für die nächste Transaktion an, wie sie durch **SET NEXTTRANSID** oder **RETURN TRANSID** festgelegt wurde. Sie gibt Leerzeichen zurück, wenn keine weiteren Transaktionen vorhanden sind.

NUMTAB(datenbereich)

Diese Option gibt eine 1-Byte-Zahl für die Tabulatoren an, die erforderlich sind, um das Druckelement im richtigen Sparsbuchbereich des 2980-Systems zu positionieren. Wenn die Task nicht von einem Terminal aus initialisiert wurde, tritt die Bedingung INVREQ auf.

OPCLASS(datenbereich)

Diese Option gibt in einer 24-Bit-Zeichenfolge die Operatorklasse zurück, die von BMS für das Routing von Terminalnachrichten verwendet wird, wie im CICS-Segment des externen Sicherheitsmanagers (External Security Manager) angegeben.

OPERATION(datenbereich)

Diese Option gibt den 64 Zeichen langen Namen der aktuellen Operation zurück, die der Task zugeordnet ist, die Teil des Anwendungskontexts ist. Wenn kein Anwendungskontext vorhanden ist, der der Task zugeordnet ist, werden Leerzeichen zurückgegeben.

OPERKEYS(datenbereich)

Diese Option wird aus Gründen der Kompatibilität mit früheren Releases akzeptiert. Wenn angegeben, wird eine 64-Bit-Nullzeichenfolge zurückgegeben.

OPID(datenbereich)

Diese Option gibt eine drei Zeichen lange Operatoridentifikation zurück. Diese Option wird von BMS zur Weiterleitung von Terminalnachrichten verwendet, wie im CICS-Segment des externen Sicherheitsmanagers (ESM) definiert.

Wenn die Task von einem fernen Terminal aus initialisiert wurde, ist die OPID, die durch diesen Befehl zurückgegeben wird, nicht unbedingt die, die dem Benutzer zugeordnet ist, der sich am fernen Terminal angemeldet hat. Wenn Sie die OPID des angemeldeten Benutzers ermitteln wollen, verwenden Sie den Systemprogrammierungsbefehl INQUIRE TERMINAL.

Die OPID kann sich außerdem dann von der OPID des Benutzers unterscheiden, der zurzeit angemeldet ist, wenn sie mit dem Befehl SET TERMINAL geändert wurde.

OPSECURITY(datenbereich)

Diese Option wird aus Gründen der Kompatibilität mit früheren Releases akzeptiert. Wenn angegeben, wird eine 24-Bit-Nullzeichenfolge zurückgegeben.

ORGABCODE(datenbereich)

Diese Option gibt in Fällen wiederholter Abbrüche den vier Byte langen ursprünglichen Abbruchcode zurück.

OUTLINE(datenbereich)

Diese Option gibt einen 1-Byte-Anzeiger zurück, der angibt, ob das Terminal mit der Feldumrandungsfunktionalität definiert ist (X'FF') oder nicht (X'00'). Wenn die Task nicht von einem Terminal aus initialisiert wurde, tritt die Bedingung INVREQ auf.

PAGENUM(datenbereich)

Diese Option gibt in einem Halbwort-Binärwert die aktuelle Seitennummer für das Ziel zurück, das einen Überlauf festgestellt hat. Wenn diese Option angegeben wird, wenn die Überlaufverarbeitung nicht aktiv ist, hat der Wert, der abgerufen wird, keine Bedeutung. Wenn keine BMS-Befehle abgesetzt werden, tritt die Bedingung INVREQ auf.

PARTNPAGE(datenbereich)

Diese Option gibt einen zwei Byte langen Namen der Partition zurück, die zuletzt einen Seitenüberlauf verursacht hat. Wenn keine BMS-Befehle abgesetzt werden, tritt die Bedingung INVREQ auf.

PARTNS(datenbereich)

Diese Option gibt einen 1-Byte-Anzeiger zurück, der zeigt, ob das Terminal Partitionen unterstützt (X'FF') oder nicht (X'00'). Wenn die Task nicht von einem Terminal aus initialisiert wurde, tritt die Bedingung INVREQ auf.

PARTNSET(datenbereich)

Diese Option gibt den Namen (1 – 6 Zeichen) der Anwendungspartitionsgruppe zurück. Ein Leerwert wird zurückgegeben, wenn keine Anwendungspartitionsgruppe vorhanden ist. Wenn die Task nicht von einem Terminal aus initialisiert wurde, tritt die Bedingung INVREQ auf.

PLATFORM(datenbereich)

Diese Option gibt den 64 Zeichen langen Namen der Plattform zurück, die der Task zugeordnet ist, die Teil des Anwendungskontexts ist. Wenn kein Anwendungskontext vorhanden ist, der der Task zugeordnet ist, werden Leerzeichen zurückgegeben.

PRINSYSID(datenbereich)

Diese Option gibt den vier Zeichen langen Namen zurück, unter dem das andere System im lokalen System bekannt ist, das heißt, den Namen der CONNECTION-Definition, die das andere System definiert. Für eine APPC-Einzelsitzungseinheit, die durch eine Terminaldefinition definiert wird, ist der zurückgegebene Wert die Terminal-ID.

Dies gilt nur, wenn es sich bei der Hauptfunktion um eine der folgenden Sitzungen handelt:

- MRO-Sitzung mit Verbindung zu einem anderen CICS-System
- LU6.1-Sitzung mit Verbindung zu einem anderen CICS- oder IMS-System
- APPC-Sitzung mit Verbindung zu einem anderen CICS-System oder zu einem anderen APPC-System bzw. zu einer anderen APPC-Einheit

Wenn die Hauptfunktion keine MRO-, LU6.1- oder APPC-Sitzung ist oder wenn die Task keine Hauptfunktion hat, tritt die Bedingung INVREQ auf.

Anmerkung: Für das Transaktionsrouting sind im Allgemeinen spezielle Hinweise zu beachten. Insbesondere kann in einer weitergeleiteten Transaktion kein Befehl **ASSIGN PRINSYSID** verwendet werden, um den Namen der Terminalverwaltungsregion (TOR - Terminal-Owning Region) zu ermitteln. Weitere Informationen finden Sie unter [CICS-Transaktionsrouting](#).

PROCESS(datenbereich)

Wenn dieses Programm für eine CICS-BTS-Aktivität (BTS - Business Transaction Services) ausgeführt wird, gibt diese Option den 36 Zeichen langen Namen des BTS-Prozesses zurück, der die Aktivität enthält.

Eine Beschreibung von BTS finden Sie in [Übersicht über BTS](#).

PROCESSTYPE(datenbereich)

Wenn dieses Programm für eine BTS-Aktivität ausgeführt wird, gibt diese Option den acht Zeichen langen Prozesstyp des BTS-Prozesses zurück, der die Aktivität enthält.

Eine Beschreibung von BTS finden Sie in [Übersicht über BTS](#).

PROGRAM(datenbereich)

Diese Option gibt einen acht Zeichen langen Namen des momentan aktiven Programms zurück.

PS(datenbereich)

Diese Option gibt einen 1-Byte-Anzeiger zurück, der angibt, ob das Terminal mit Funktionalität für programmierte Symbole definiert ist (X'FF') oder nicht (X'00'). Wenn die Task nicht von einem Terminal aus initialisiert wurde, tritt die Bedingung INVREQ auf.

QNAME(datenbereich)

Diese Option gibt einen vier Zeichen langen Namen für die partitionsinterne Warteschlange für transiente Daten zurück, die die Initialisierung dieser Task durch Erreichen der Auslöserebene verursacht hat. Wenn die Task nicht durch die automatische Transaktionsinitialisierung (ATI) initialisiert wurde, tritt die Bedingung INVREQ auf.

RESSEC(datenbereich)

Diese Option gibt einen 1-Byte-Anzeiger zurück, der angibt, ob für die Transaktion, die ausgeführt wird, die Ressourcensicherheitsprüfung definiert ist. (X für ja, leer für nein.)

RESTART(datenbereich)

Diese Option gibt einen 1-Byte-Anzeiger zurück, der angibt, ob ein Neustart der Task (X'FF') oder ein normaler Start der Task (X'00') stattgefunden hat.

RETURNPROG(datenbereich)

Diese Option gibt den acht Zeichen langen Namen des Programms an, an das die Steuerung zurückgegeben werden soll, wenn die Ausführung des aktuellen Programms abgeschlossen ist. Die Werte, die zurückgegeben werden, hängen wie folgt davon ab, wie die Steuerung an das aktuelle Programm übergeben wurde:

- Wenn das aktuelle Programm durch einen Befehl **LINK** (einschließlich Distributed Program Link) aufgerufen wurde, gibt die Option RETURNPROG denselben Namen wie die Option INVOKINGPROG zurück.
- Wenn das aktuelle Programm durch einen Befehl **XCTL** aufgerufen wurde, gibt die Option RETURNPROG den Namen des Anwendungsprogramms in der Kette zurück, das zuletzt einen Befehl LINK abgesetzt hat.

Wenn sich das Programm, das das aktuelle Programm mit einem Befehl **XCTL** aufgerufen hat, auf der höchsten Ebene befindet, gibt CICS acht Leerzeichen zurück.

- Wenn der Befehl **ASSIGN RETURNPROG** in dem Programm auf der höchsten Ebene abgesetzt wird, gibt CICS acht Leerzeichen zurück.
- Wenn der Befehl **ASSIGN RETURNPROG** in einem durch den Benutzer austauschbaren Modul oder in einem Programmllistentabellenprogramm abgesetzt wird, gibt CICS acht Leerzeichen zurück.
- Wenn der Befehl **ASSIGN RETURNPROG** in einem globalen Benutzerexit, einem taskbezogenen Exit oder in einem Anwendungsprogramm, das über einen solchen Exit verknüpft wurde, abgesetzt wird,

gibt CICS den Namen des Programms zurück, an das die Steuerung zurückgegeben wird, wenn alle zwischengeschalteten Programme globaler oder taskbezogener Benutzerexits abgeschlossen sind.

Beispiel:

```
Programm A stellt Verbindung zu Programm B her.  
Programm B stellt Verbindung zu Programm C her.  
Programm C übergibt die Steuerung durch XCTL an Programm D.  
Programm D setzt Befehl ASSIGN RETURNPROG ab und CICS gibt den Namen von  
Programm B zurück.
```

SCRNHT(datenbereich)

Diese Option gibt eine Halbwort-Binärwert-Variable zurück, die die Höhe der 3270-Anzeige enthält, die für die aktuelle Task definiert ist. Wenn die Task nicht von einem Terminal aus initialisiert wurde, tritt die Bedingung INVREQ auf.

SCRNWD(datenbereich)

Diese Option gibt eine Halbwort-Binärwert-Variable zurück, die die Breite der 3270-Anzeige enthält, die für die aktuelle Task definiert ist. Wenn die Task nicht von einem Terminal aus initialisiert wurde, tritt die Bedingung INVREQ auf.

SIGDATA(datenbereich)

Diese Option gibt eine vier Byte lange Zeichenfolge zurück, die die eingehenden Signaldaten enthält, die von einer logischen Einheit empfangen wurden. Wenn die Task nicht von einem Terminal aus initialisiert wurde, tritt die Bedingung INVREQ auf.

SOSI(datenbereich)

Diese Option gibt einen 1-Byte-Anzeiger zurück, der angibt, ob das Terminal mit Funktionalität für gemischte EBCDIC/DBCS-Felder definiert ist (X'FF') oder nicht (X'00'). Die DBCS-Unterfelder in einem EBCDIC-Feld werden durch ein DBCS-Startzeichen (SO - Shift-out) und ein DBCS-Endezeichen (SI - Shift-in) begrenzt. Wenn die Task nicht von einem Terminal aus initialisiert wurde, tritt die Bedingung INVREQ auf.

STARTCODE(datenbereich)

Diese Option gibt einen Wert aus zwei Zeichen zurück, der angibt, wie die Transaktion gestartet wurde, die die Anforderung abgesetzt hat. Die folgenden Werte sind möglich:

Code

Transaktion gestartet durch

D

Eine Distributed Program Link-Anforderung (DPL-Anforderung), in der die Option SYNCONRETURN nicht angegeben wurde. Die Task kann keine Ein-/Ausgabeanforderungen für ihre Hauptfunktion und keine Synchronisationspunktanforderungen absetzen.

DS

Eine DPL-Anforderung wie unter Code D, in der die Option SYNCONRETURN angegeben wurde. Die Task kann Synchronisationspunktanforderungen absetzen.

QD

Auslöserebene für transiente Daten.

S

Befehl START, der keine Daten in der Option FROM übergeben hat. Der Befehl kann einen Kanal übergeben haben oder auch nicht.

SD

Befehl START, der Daten in der Option FROM übergeben hat.

SZ

Befehl **FEPI START**.

TD

Terminaleingabe oder permanente Transaktions-ID (TRANSID).

U

Benutzerangeschlossene Task.

STATIONID(datenbereich)

Diese Option gibt eine 1 Byte große Stations-ID eines 2980-Systems zurück. Wenn die Task nicht von einem Terminal aus initialisiert wurde, tritt die Bedingung INVREQ auf.

SYSID(datenbereich)

Diese Option gibt den vier Zeichen langen Namen zurück, der dem lokalen CICS-System gegeben wurde. Dieser Wert kann in der Option SYSID eines Befehls für Dateisteuerung, Intervallsteuerung, temporären Speicher oder transiente Daten angegeben werden. In diesem Fall wird davon ausgegangen, dass sich die Ressource, auf die zugegriffen werden soll, auf dem lokalen System befindet.

TASKPRIORITY(datenbereich)

Diese Option gibt ein Halbwort-Binärfeld zurück, das die aktuelle Priorität der absetzenden Task (0 – 255) angibt. Wenn die Task zu Anfang angeschlossen wird, ist dies die Summe der Prioritäten für den Benutzer, das Terminal und die Transaktion. Dieser Wert kann während der Ausführung durch einen Befehl **CHANGE TASK** geändert werden.

TCTUALENG(datenbereich)

Diese Option gibt einen Halbwort-Binärwert für die Länge des Benutzerbereichs der Terminalsteuer-tabelle (Terminal Control Table User Area - TCTUA) zurück. Wenn kein TCTUA vorhanden ist, wird die Länge null zurückgegeben.

TELLERID(datenbereich)

Diese Option gibt eine 1 Byte große Bankautomaten-ID (Teller) eines 2980-Systems zurück. Wenn die Task nicht von einem Terminal aus initialisiert wurde, tritt die Bedingung INVREQ auf.

TERMCODE(datenbereich)

Diese Option gibt einen zwei Byte langen Code zurück, der den Typ und die Modellnummer des Terminals angibt, das der Task zugeordnet ist.

Das erste Byte ist ein Code, der den Terminaltyp identifiziert, der aus der TERMINAL-Ressource abgeleitet wird. Eine Beschreibung der Ressourcenattribute finden Sie unter [TERMINAL-Attribute](#). Das zweite Byte enthält ein einzelnes Zeichen für eine Modellnummer, wie sie im Attribut TERMMODEL angegeben ist.

Die Bedeutungen der Typcodes finden Sie unter [Von ASSIGN zurückgegebene Codes](#).

TERMPRIORITY(datenbereich)

Diese Option gibt einen Halbwort-Binärwert für die Terminalpriorität (0 – 255) zurück.

TEXTKYBD(datenbereich)

Diese Option gibt einen 1-Byte-Anzeiger zurück, der angibt, ob die Hauptfunktion TEXTKYBD unterstützt (X'FF') oder nicht (X'00'). Wenn die Task nicht von einem Terminal aus initialisiert wurde, tritt die Bedingung INVREQ auf.

TEXTPRINT(datenbereich)

Diese Option gibt einen 1-Byte-Anzeiger zurück, der angibt, ob die Hauptfunktion TEXTPRINT unterstützt (X'FF') oder nicht (X'00'). Wenn die Task nicht von einem Terminal aus initialisiert wurde, tritt die Bedingung INVREQ auf.

TNADDR(datenbereich)

Gibt einen 39 Zeichen langen Bereich mit der IPv4- oder IPv6-Adresse des TN3270-Clients zurück. Wenn TNIPFAMILY den Wert NOTAPPLIC zurückgibt, gibt TNADDR Leerzeichen zurück.

Wenn die Task nicht von einem Terminal aus initialisiert wurde, tritt die Bedingung INVREQ auf.

TNIPFAMILY(cvda)

Gibt das Adressformat der Option TNADDR zurück. Die folgenden CVDA-Werte sind möglich:

NOTAPPLIC

Dieser Wert gibt eine der folgenden Bedingungen an:

- Das Terminal ist keine 3270-Einheit.
- TNADDR wird nicht verwendet.
- Die Adresse kann nicht aufgelöst werden.

IPV4

Die Option TNADDR enthält eine IPv4-Adresse in der Schreibweise mit Trennzeichen.

IPV6

Die Option TNADDR enthält eine IPv6-Hexadezimaladresse in Doppelpunktschreibweise.

Wenn die Task nicht von einem Terminal aus initialisiert wurde, tritt die Bedingung INVREQ auf.

TNPORT(datenbereich)

Gibt einen Vollwort-Binärwert zurück, der die Portnummer enthält, die für die TN3270-Clientverbindung verwendet wird. Falls das Terminal keine 3270-Einheit ist, gibt die Option TNPORT den Wert null zurück.

Wenn die Task nicht von einem Terminal aus initialisiert wurde, tritt die Bedingung INVREQ auf.

TRANPRIORITY(datenbereich)

Diese Option gibt einen Halbwort-Binärwert für die Transaktionspriorität (0 – 255) zurück.

TWALENG(datenbereich)

Diese Option gibt einen Halbwort-Binärwert für die Länge des Transaktionsarbeitsbereichs (TWA) zurück. Wenn kein TWA vorhanden ist, wird die Länge null zurückgegeben.

UNATTEND(datenbereich)

Diese Option gibt einen 1-Byte-Anzeiger zurück, der angibt, ob die Betriebsart des Terminals unbeaufsichtigt ist, das heißt, dass keine Person das Terminal bedient. Diese Anzeiger sind X'FF' für 'unbeaufsichtigt' und X'00' 'beaufsichtigt'. Wenn die Task nicht von einem Terminal aus initialisiert wurde, tritt die Bedingung INVREQ auf.

USERID(datenbereich)

Diese Option gibt eine 8 Byte lange Benutzer-ID des angemeldeten Benutzers zurück. Wenn keine Benutzer explizit angemeldet ist, gibt CICS die Standard-Benutzer-ID zurück. Besondere Aspekte sind zu berücksichtigen, wenn Sie eine Umgebung für übergreifende Kommunikation verwenden. Weitere Informationen zum Befehl **ASSIGN** für LUTYPE6.1, APPC und MRO finden Sie in [Erste Schritte mit übergreifender Kommunikation](#).

USERNAME(datenbereich)

Diese Option gibt einen 20 Zeichen langen Namen des Benutzers zurück, der aus dem externen Sicherheitsmanager (ESM) abgerufen wurde.

USERPRIORITY(datenbereich)

Diese Option gibt einen Halbwort-Binärwert für die Operatorpriorität (0 – 255) zurück.

VALIDATION(datenbereich)

Diese Option gibt einen 1-Byte-Anzeiger zurück, der angibt, ob das Terminal mit Prüffunktionalität definiert ist (X'FF') oder nicht (X'00'). Die Prüffunktionalität besteht aus den Attributen für obligatorisches Füllen, obligatorische Eingabe und Auslöser (Trigger). Wenn die Task nicht von einem Terminal aus initialisiert wurde, tritt die Bedingung INVREQ auf.

Bedingungen

Der Befehl **ASSIGN** gibt unter CECI oder in einem REXX-Programm immer die Ausnahmebedingung INVREQ zurück. Auch wenn CECI oder das REXX-Programm möglicherweise Informationen zurückgeben, die Sie korrekt angefordert haben, wird außerdem versucht, Informationen aus anderen Optionen abzurufen, von denen einige ungültig sind.

16 INVREQ

RESP2-Werte:

2

Es wurde noch kein BMS-Befehl abgesetzt, das BMS-Routing ist aktiv oder es wurde keine Maske positioniert.

3

Es wurde noch kein Stapeldatenaustauschbefehl (BDI - Batch Data Interchange) abgesetzt.

4

Die Task wurde nicht durch die automatische Transaktionsinitialisierung (ATI) initialisiert.

5

Die Task ist keinem Terminal zugeordnet, die Task hat keine Hauptfunktion oder die Hauptfunktion ist keine MRO-, LU6.1- oder APPC-Sitzung.

6

Eine CICS-BTS-Anforderung wurde von außerhalb der CICS-BTS-Umgebung abgesetzt. Daher wird die Transaktion nicht für eine BTS-Aktivität ausgeführt.

200

Befehlssyntaxoptionen sind in einem Serverprogramm nicht zulässig, das durch einen Distributed Program Link aufgerufen wird.

Standardaktion: Die Task wird abnormal beendet.

Von ASSIGN zurückgegebene Codes

Der Befehl **ASSIGN** kann Codes in den Optionen TERMCODE und FCI zurückgeben. Das Format und die Bedeutung der Codes werden beschrieben.

ASSIGN TERMCODE

In der folgenden Liste sind die Bedeutungen der Terminaltypcodes aufgeführt, die durch die Option TERMCODE des Befehls ASSIGN im ersten Byte des Datenbereichs zurückgegeben werden.

Die Codes werden aus dem Attribut DEVICE der TYPETERM-Ressource abgeleitet. Das zweite Byte des Datenbereichs enthält eine Modellnummer in Zeichenform, die durch das Attribut TERMMODEL der TYPE-TERM-Ressourcendefinition festgelegt ist.

Die Codes sind hier als Bitmuster und auch als Hexadezimalwerte aufgeführt.

Code		Bedeutung
.... .1	X'01'	7770
.... .1.	X'02'	System 7
.... 1...	X'08'	Konsole
...1 .1.	X'12'	Sequenzielle Platte
...1 .1..	X'14'	Magnetband
...1 1...	X'18'	Kartenleser oder Zeilendrucker
...1 1..1	X'19'	Spoolsystemdrucker
...1 1.1.	X'1A'	Interner Leser für Spoolbetrieb
..1.	X'20'	Hardcopy-Terminals
..1. ...1	X'21'	Modell 33/35 TWX
..1. .1.	X'22'	Fernschreiber
..1. .1..	X'24'	1050
..1. 1...	X'28'	2740
..1. 1.1.	X'2A'	2741-Entsprechung
..1. 1.11	X'2B'	2741-EBCDIC
.1..	X'40'	Videoterminals
.1.. ...1	X'41'	2260 (lokal)
.1.. 1...	X'48'	2260 (fern)
.1.. 1.1.	X'4A'	1053
.1.. 11..	X'4C'	2265
1...	X'80'	Bisynchron

Code		Bedeutung
1... ..1.	X'82'	2770
1... .1..	X'84'	2780
1... .1.1	X'85'	3780
1... .11.	X'86'	2980
1... 1...	X'88'	3735
1... 1..1	X'89'	3740
1... 1.1.	X'8A'	3600 (bisynchron)
1..1 ...1	X'91'	3277 (fern)
1..1 ..1.	X'92'	3275 (fern)
1..1 1..1	X'99'	3277 (lokal)
1.1.	X'A0'	Bisynchron (programmierbar)
1.1. ...1	X'A1'	System/3
1.1. .1..	X'A4'	z Systems
1.1. .11.	X'A6'	System/7 mit BSCA
1.11	X'B0'	SDLC-Einheitenklasse
1.11 ...1	X'B1'	3601
1.11 ..1.	X'B2'	3614
1.11 .1..	X'B4'	3790
1.11 .1.1	X'B5'	3790-Benutzerprogramm
1.11 .11.	X'B6'	3790-SCS-Drucker
1.11 1...	X'B8'	3650-Pipeline
1.11 1..1	X'B9'	3653-Hostdialog
1.11 1.1.	X'BA'	3650 mit angeschlossenem 3270-Hostdialog
1.11 1.11	X'BB'	3650-Benutzerprogramm
1.11 11.1	X'BD'	Logische Einheit für Konkurrenzsituation
1.11 111.	X'BE'	Interaktive logische Einheit
1.11 1111	X'BF'	Logische Einheit mit Stapel
11..	X'C0'	LU-Typ 6 Anmerkung: Ein Befehl ASSIGN TERMCODE für eine ISC-Sitzung gibt für den LU-Typ 6 X'C0' zurück. Anschließend bestimmt ein Befehl INQUIRE CONNECTION, ob diese ISC-Verbindung LU-Typ 6.1 oder APPC-Protokolle verwendet.
11.. ...1	X'C1'	LU-Typ 4
11.1 ...1	X'D1'	ISC-MM-Dialog
11.1 ..1.	X'D2'	LUC-Modusgruppeneintrag
11.1 ..11	X'D3'	LUC-Sitzung

Option FCI des Befehls ASSIGN

In der folgenden Liste sind die Bedeutungen der Codes für den Funktionssteueranzeiger in dem Datenbereich aufgeführt, der durch die Option FCI des Befehls ASSIGN zurückgegeben wird.

Code		Bedeutung
....	X'00'	Keine
.... ...1	X'01'	Anschlussfunktionsanzeiger Dieser Code gibt an, dass die Hauptfunktion für die Task ein Terminal ist.
.... ..1.	X'02'	KCP-Makrodateimaske Dieser Code wird nicht mehr verwendet.
.... .1..	X'04'	Intervallsteuerungsanzeiger Dieser Code gibt an, dass die Hauptfunktion für die Task ein Intervallsteuerelement (Interval Control Element, ICE) ist.
.... 1...	X'08'	Zielsteueranzeiger Dieser Code gibt an, dass die Hauptfunktion für die Task eine Warteschlange mit transienten Daten ist.
...1	X'10'	AID-Funktionsmaske Dieser Code wird nicht mehr verwendet.
111.	X'E0'	Reserviert

BIF DEEDIT

Stellt die Funktion DEEDIT bereit, die Sonderzeichen aus einem EBCDIC-Datenfeld entfernt.

BIF DEEDIT

➡ BIF DEEDIT — FIELD(*datenbereich*) ————> |
|—————|
| LENGTH(*datenwert*) |

Bedingung: LENGERR

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl BIF DEEDIT stellt die integrierte Funktion DEEDIT bereit. Er gibt an, dass Buchstaben und Sonderzeichen aus dem EBCDIC-Datenfeld entfernt werden und dass die verbleibenden Ziffern rechtsbündig ausgerichtet und bei Bedarf auf der linken Seite mit Nullen aufgefüllt werden.

Wenn das Feld mit einem Minuszeichen oder einem Wagenrücklaufzeichen (CR) endet, wird eine negative Zone (X'D') in das ganz rechte Byte (niedrigstwertiges Byte) eingefügt.

Wenn der Zonenabschnitt des ganz rechten Byte eines der Zeichen X'A' bis X'F' enthält und der numerische Abschnitt eine der Hexadezimalziffern X'0' bis X'9' enthält, wird das ganz rechte Byte ungeändert zurückgegeben (siehe Beispiel). Dies bietet dem Anwendungsprogramm die Möglichkeit, auf einem gezonten numerischen Feld zu operieren. Der zurückgegebene Wert ist das Feld, das ursprünglich die unearbeiteten Daten enthielt.

Ein 1-Byte-Feld wird unabhängig vom Inhalt des Felds ungeändert zurückgeben.

Optionen

FIELD(*datenbereich*)

Diese Option gibt das Feld an, das zu bearbeiten ist.

LENGTH(*datenwert*)

Diese Option gibt die Feldlänge in Byte an.

Bedingungen

22 LENGERR

Diese Bedingung tritt auf, wenn der Wert von LENGTH kleiner als 1 ist.

Standardaktion: Die Task wird abnormal beendet.

Beispiele

```
EXEC CICS BIF DEEDIT  
      FIELD(CONTG)  
      LENGTH(9)
```

Dieser Befehl entfernt alle Zeichen außer Ziffern aus dem neun Byte langen Feld CONTG und gibt das bearbeitete Ergebnis in diesem Feld an das Anwendungsprogramm zurück.

Die folgenden beiden Beispiele zeigen den Inhalt des Felds CONTG vor und nach Ausführung des Befehls:

Originalwert	Rückgabewert
14-6704/B	00146704B
\$25.68	000002568

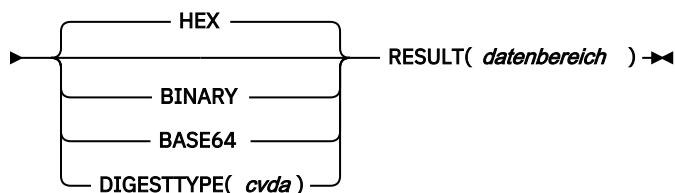
Ein Dezimaltrennzeichen ist ein EBCDIC-Sonderzeichen und wird als solches entfernt.

BIF DIGEST

Berechnet den SHA-1-Auszug einer Datenzeichenfolge.

BIF DIGEST

➡ BIF — DIGEST — RECORD(*datenwert*) — RECORDLEN(*datenwert*) — ➡



Bedingungen: INVREQ, LENGERR

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl **BIF DIGEST** ist eine integrierte CICS-Funktion, die den SHA-1-Auszug (Digest) einer Datenzeichenfolge berechnet. Das Ergebnis kann als Binärwert (20 Byte lang), als Hexadezimalwert (40 Byte lang) oder als Wert in Base64-Codierung (28 Byte lang) zurückgegeben werden. Der SHA-1-Auszug ist eine kryptografisch starke Kontrollsumme der Zeichenfolge, die für jede Zeichenfolge praktisch eindeutig ist.

Dieser Befehl nutzt MSA-Funktionen (Message Security Assist) von z/Architecture, für die z Systems-Verschlüsselungshardware mit CP-Assist für Verschlüsselungsfunktionen (CP Assist for Cryptographic Func-

tions - CPACF) erforderlich ist. Weitere Informationen finden Sie unter [z/OS Cryptographic Services ICSF Overview](#).

Optionen

RECORD(datenwert)

Diese Option gibt die Datenzeichenfolge an, für die der Auszug (Digest) berechnet werden soll.

RECORDLEN(datenwert)

Diese Option gibt die Länge der Datenzeichenfolge als Vollwort-Binärwert an.

DIGESTTYPE(cvda)

Diese Option gibt das Format an, in dem der Auszug zurückgegeben wird.

HEX

Hexadezimal: Generiert ein 40 Byte langes Ergebnis, das in Hexadezimalzeichen (0 - 9, A - F) codiert ist.

BINARY

Binär: Generiert ein 20 Byte langes Ergebnis.

BASE64

Base64-Codierung: Generiert ein 28 Byte langes Ergebnis unter Verwendung der Zeichen A - Z, a - z, 0 - 9, +, /, =.

RESULT(datenbereich)

Diese Option gibt den SHA-1-Auszug der Datenzeichenfolge in dem Format zurück, das durch die Option DIGESTTYPE angegeben wird. Die Länge des Ergebnisses hängt vom angeforderten Format ab.

Bedingungen

16 INVREQ

RESP2-Werte:

1

DIGESTTYPE hat einen ungültigen CVDA-Wert.

3

z/Architecture Message Security Assist (MSA) ist nicht verfügbar.

22 LENGERR

RESP2-Werte:

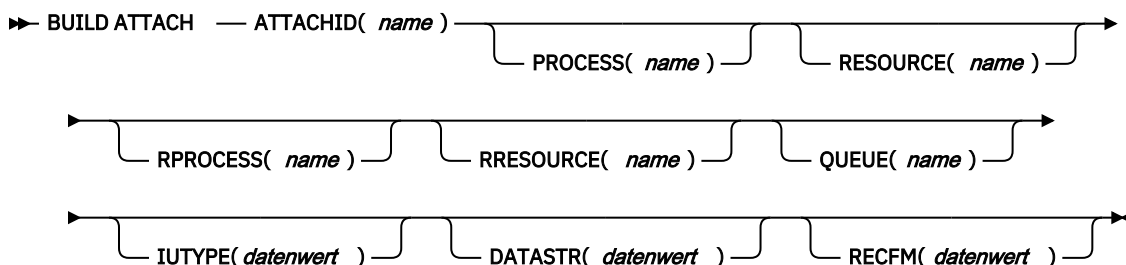
2

Der Wert für RECORDLEN ist kleiner als 1.

BUILD ATTACH (LUTYPE6.1)

Gibt Werte für einen LUTYPE6.1-Anschlussheader an.

BUILD ATTACH (LUTYPE6.1)



Beschreibung

Der Befehl BUILD ATTACH (LUTYPE6.1) gibt eine Gruppe von Werten an, die in den benannten Anschlussheadersteuerblock (Attach Header Control Block) eingefügt werden sollen. Dieser Steuerblock enthält

Werte, die in einem LUTYPE6.1-Funktionsverwaltungsheder (FMH) für den Anschluss gesendet werden sollen, der von CICS erstellt wird und der nur gesendet wird, wenn ein Befehl SEND ATTACHID oder CONVERSE ATTACHID ausgeführt wird. Die angegebenen Werte überschreiben vorhandene Werte in dem Steuerblock. Nicht angegebene Werte werden auf Standardwerte gesetzt.

Optionen

ATTACHID(name)

Diese Option gibt an, dass die Gruppe von Werten in einen Steuerblock für einen Anschlussheader eingefügt werden soll, der durch den angegebenen Namen (1 - 8 Zeichen) identifiziert wird.

DATASTR(datenwert)

Diese Option entspricht dem Datenstromprofilfeld ATTDSP in einem Anschlussfunktionsverwaltungsheder für LUTYPE6.1.

Für die Kommunikation zwischen zwei CICS-Systemen hat das Datenstromprofilfeld in einem Anschlussfunktionsverwaltungsheder für CICS keine besondere Bedeutung. Für die meisten CICS-Anwendungen kann die Option weggelassen werden.

Weitere Informationen zur Kommunikation zwischen einem CICS-System und einem beliebigen anderen Subsystem, einschließlich Details zu strukturierten Feldern und zur Verwaltung logischer Datensätze, finden Sie in der für das Subsystem bereitgestellten Dokumentation zur Verwendung des Datenstromprofilfelds in einem Anschlussfunktionsverwaltungsheder (FMH).

Der "datenwert" ist ein Halbwort-Binärwert. Nur das niedrigstwertige Byte wird verwendet. Die SNA-definierten Bedeutungen der Bits sind folgende:

0-7	Reserviert - müssen auf null gesetzt werden.
8-11	0000 - Benutzerdefiniert.
	1111 - SCS-Datenstrom.
	1110 - 3270-Datenstrom.
	1101 - Strukturiertes Feld.
	1100 - Verwaltung logischer Datensätze.
12-15	Durch Benutzer definiert, wenn Bits 8-11 auf 0000 gesetzt sind; ansonsten reserviert. (Müssen auf null gesetzt werden.)

Der Wert "Strukturiertes Feld" gibt an, dass Ketten mit vier Byte von Daten beginnen, die zum Interpretieren der folgenden Daten verwendet werden: Gesamtlänge (2 Byte), Klassen-ID (1 Byte) und Unterklassen-ID (1 Byte). Der Wert "Verwaltung logischer Datensätze" gibt an, dass Ketten vom Datenempfänger in separate Felder aufgliedert werden können.

Wenn die Option nicht angegeben wird, wird der Wert "Benutzerdefiniert" angenommen.

IUTYPE(datenwert)

Diese Option entspricht dem Austauschseinheitenfeld ATTU in einem Anschlussfunktionsverwaltungsheder für LUTYPE6.1.

Der "datenwert" ist ein Halbwort-Binärwert. Nur die 7 niedrigstwertigen Byte werden verwendet. Die SNA-definierten Bedeutungen der Bits sind folgende:

0-10	Reserviert - müssen auf null gesetzt werden.
11	0 - Nicht Ende der Mehrfachketten-Austauscheinheit
	1 - Ende der Mehrfachketten-Austauscheinheit
12,13	Reserviert - müssen auf null gesetzt werden.
14,15	00 - Mehrfachketten-Austauscheinheit
	01 - Einzelketten-Austauscheinheit
	10 - Reserviert.
	11 - Reserviert.

Wenn die Option nicht angegeben wird, werden die Werte "Nicht Ende der Mehrfachketten-Austauscheinheit" und "Mehrfachketten-Austauscheinheit" angenommen.

PROCESS(name)

Diese Option entspricht dem Prozessnamen ATTDPN in einem Anschlussfunktionsverwaltungsheader für LUTYPE6.1.

Für die Kommunikation zwischen zwei CICS-Systemen kann eine Transaktion, die in dem einen System ausgeführt wird, eine Sitzung zu dem zweiten System anfordern und die Transaktion angeben, die angeschlossen (Attach) werden soll. In dem zweiten System wird die Identifikation in der ersten Kette von Daten transportiert, die über die Sitzung gesendet wird.

Im Allgemeinen identifizieren die ersten vier Byte von Daten die anzuschließende Transaktion. Jedoch kann ein Anschlussfunktionsverwaltungsheader (FHM), der die anzuschließende Transaktion identifiziert, erstellt und gesendet werden. Die Option PROCESS wird verwendet, um den Transaktionsnamen anzugeben. (Das empfangende CICS-System verwendet nur die ersten vier Byte des Prozessnamens als Transaktionsnamen.)

Von CICS wird Prozessnamen in Anschlussfunktionsverwaltungsheadern, die in anderen Ketten als der ersten Kette von Daten gesendet werden, keine Bedeutung beigemessen.

Informationen zur Kommunikation zwischen einem CICS-System und einem anderen Subsystem finden Sie in der für das Subsystem bereitgestellten Dokumentation zur Verwendung des Prozessnamensfelds in einem Anschlussfunktionsverwaltungsheader (FMH).

QUEUE(name)

Diese Option entspricht dem Warteschlangennamen ATTDQN in einem Anschlussfunktionsverwaltungsheader für LUTYPE6.1.

Für die Kommunikation zwischen zwei CICS-Systemen hat der Warteschlangenname in einem Anschlussfunktionsverwaltungsheader für CICS keine Bedeutung.

Informationen zur Kommunikation zwischen einem CICS-System und einem anderen Subsystem finden Sie in der für das Subsystem bereitgestellten Dokumentation zur Verwendung des Warteschlangennamensfelds in einem Anschlussfunktionsverwaltungsheader (FMH).

RECFM(datenwert)

Diese Option entspricht dem Feld für den Entblockungsalgorithmus ATTDDBA in einem Anschlussfunktionsverwaltungsheader für LUTYPE6.1.

Für die Kommunikation zwischen zwei CICS-Systemen hat das Feld für den Entblockungsalgorithmus in einem Anschlussfunktionsverwaltungsheader für CICS keine besondere Bedeutung. Für die meisten CICS-Anwendungen kann die Option weggelassen werden.

Der "datenwert" ist ein Halbwort-Binärwert. Nur das niedrigstwertige Byte wird verwendet. Die SNA-definierten Bedeutungen der Bits sind folgende:

0-7	Reserviert - müssen auf null gesetzt werden.
8-15	X'00' - Reserviert.
	X'01' - Variable Länge.
	Variable geblockt.
	X'02' - Reserviert.
	X'03' - Reserviert.
	X'04' - Kette von RUs (Anforderung/Antwort-Einheiten).
	X'05' bis X'FF' - Reserviert.

Wenn die Option nicht angegeben wird, wird der Wert "Kette von RUs" angenommen.

RESOURCE(name)

Diese Option entspricht dem Ressourcennamen ATTPRN in einem Anschlussfunktionsverwaltungsheader für LUTYPE6.1.

RPROCESS(name)

Diese Option entspricht dem Rückgabeprozessnamen ATTRDPN in einem Anschlussfunktionsverwaltungsheader für LUTYPE6.1.

Für die Kommunikation zwischen zwei CICS-Systemen hat der Rückgabeprozessname in einem Anschlussfunktionsverwaltungsheader für CICS keine Bedeutung.

Informationen zur Kommunikation zwischen einem CICS-System und einem anderen Subsystem finden Sie in der für das Subsystem bereitgestellten Dokumentation zur Verwendung des Felds für den Rückgabeprozessnamen in einem Anschlussfunktionsverwaltungsheader (FMH).

RRESOURCE(name)

Diese Option entspricht dem Rückgabereessourcennamen ATTRPRN in einem Anschlussfunktionsverwaltungsheader für LUTYPE6.1.

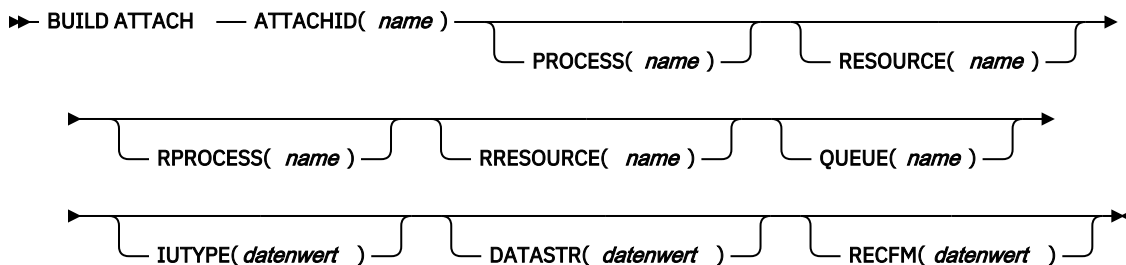
Für die Kommunikation zwischen zwei CICS-Systemen hat der Rückgabereessourcenname in einem Anschlussfunktionsverwaltungsheader für CICS keine Bedeutung.

Informationen zur Kommunikation zwischen einem CICS-System und einem anderen Subsystem finden Sie in der für das Subsystem bereitgestellten Dokumentation zur Verwendung des Felds für den Rückgabereessourcennamen in einem Anschlussfunktionsverwaltungsheader (FMH).

BUILD ATTACH (MRO)

Gibt Werte für einen MRO-Anschlussheader an.

BUILD ATTACH (MRO)



Beschreibung

Der Befehl BUILD ATTACH (MRO) gibt eine Gruppe von Werten an, die in den benannten Anschlussheadersteuerblock (Attach Header Control Block) eingefügt werden sollen. Dieser Steuerblock enthält Werte, die in einem MRO-Funktionsverwaltungsheader (FMH) für den Anschluss gesendet werden sollen, der von CICS erstellt wird und der nur gesendet wird, wenn ein Befehl SEND ATTACHID oder CONVERSE ATTACHID ausgeführt wird. Die angegebenen Werte überschreiben vorhandene Werte in dem Steuerblock. Nicht angegebene Werte werden auf Standardwerte gesetzt.

Weitere Informationen zu MRO und IRC finden Sie unter [Einführung in die übergreifende Kommunikation in CICS](#).

Optionen

ATTACHID(name)

Diese Option gibt an, dass die Gruppe von Werten in einen Steuerblock für einen Anschlussheader eingefügt werden soll, der durch den angegebenen Namen (1-8 Zeichen) identifiziert wird.

DATASTR(datenwert)

Diese Option entspricht dem Datenstromprofilfeld ATTDSP in einem Anschlussfunktionsverwaltungsheader für LUTYPE6.1.

Für die Kommunikation zwischen zwei CICS-Systemen hat das Datenstromprofilfeld in einem Anschlussfunktionsverwaltungsheader für CICS keine besondere Bedeutung. Für die meisten CICS-Anwendungen kann die Option weggelassen werden.

Der "datenwert" ist ein Halbwort-Binärwert. Nur das niedrigstwertige Byte wird verwendet. Die SNA-definierten Bedeutungen der Bits sind folgende:

0-7	Reserviert - müssen auf null gesetzt werden.
8-11	0000 - Benutzerdefiniert. 1111 - SCS-Datenstrom. 1110 - 3270-Datenstrom. 1101 - Strukturiertes Feld. 1100 - Verwaltung logischer Datensätze.
12-15	Durch Benutzer definiert, wenn Bits 8-11 auf 0000 gesetzt sind; ansonsten reserviert. (Müssen auf null gesetzt werden.)

Der Wert “Strukturiertes Feld” gibt an, dass Ketten mit vier Byte von Daten beginnen, die zum Interpretieren der folgenden Daten verwendet werden: Gesamtlänge (2 Byte), Klassen-ID (1 Byte) und Unterklassen-ID (1 Byte). Der Wert “Verwaltung logischer Datensätze” gibt an, dass Ketten vom Datenempfänger in separate Felder aufgliedert werden können.

Wenn die Option nicht angegeben wird, wird der Wert “Benutzerdefiniert” angenommen.

IUTYPE(datenwert)

Diese Option entspricht dem Austauschheitenfeld ATTIU in einem Anschlussfunktionsverwaltungsheader für LUTYPE6.1.

Der “datenwert” ist ein Halbwort-Binärwert. Nur die 7 niedrigstwertigen Byte werden verwendet. Die SNA-definierten Bedeutungen der Bits sind folgende:

0-10	Reserviert - müssen auf null gesetzt werden.
11	0 - Nicht Ende der Mehrfachketten-Austauscheinheit. 1 - Ende der Mehrfachketten-Austauscheinheit.
12,13	Reserviert - müssen auf null gesetzt werden.
14,15	00 - Mehrfachketten-Austauscheinheit. 01 - Einzelketten-Austauscheinheit. 10 - Reserviert. 11 - Reserviert.

Wenn die Option nicht angegeben wird, werden die Werte “Nicht Ende der Mehrfachketten-Austauscheinheit” und “Mehrfachketten-Austauscheinheit” angenommen.

PROCESS(name)

Diese Option entspricht dem Prozessnamen ATTDPN in einem Anschlussfunktionsverwaltungsheader für LUTYPE6.1.

Für die Kommunikation zwischen zwei CICS-Systemen kann eine Transaktion, die in dem einen System ausgeführt wird, eine Sitzung zu dem zweiten System anfordern und die Transaktion angeben, die angeschlossen (Attach) werden soll. In dem zweiten System wird die Identifikation in der ersten Kette von Daten transportiert, die über die Sitzung gesendet wird. Im Allgemeinen identifizieren die ersten vier Byte von Daten die anzuschließende Transaktion. Jedoch kann ein Anschlussfunktionsverwaltungsheader (FHM), der die anzuschließende Transaktion identifiziert, erstellt und gesendet werden. Die Option PROCESS wird verwendet, um den Transaktionsnamen anzugeben. (Das empfangende CICS-System verwendet nur die ersten vier Byte des Prozessnamens als Transaktionsnamen.)

Von CICS wird Prozessnamen in Anschlussfunktionsverwaltungsheadern, die in anderen Ketten als der ersten Kette von Daten gesendet werden, keine Bedeutung beigemessen.

Informationen zur Kommunikation zwischen einem CICS-System und einem anderen Subsystem finden Sie in der für das Subsystem bereitgestellten Dokumentation zur Verwendung des Prozessnamensfelds in einem Anschlussfunktionsverwaltungsheader (FMH).

QUEUE(name)

Diese Option entspricht dem Warteschlangennamen ATTDQN in einem Anschlussfunktionsverwaltungsheader.

Für die Kommunikation zwischen zwei CICS-Systemen hat der Warteschlangenname in einem Anschlussfunktionsverwaltungsheader für CICS keine Bedeutung.

Informationen zur Kommunikation zwischen einem CICS-System und einem anderen Subsystem finden Sie in der für das Subsystem bereitgestellten Dokumentation zur Verwendung des Warteschlangennamensfelds in einem Anschlussfunktionsverwaltungsheader (FMH).

RECFM(*datenwert*)

Diese Option entspricht dem Feld für den Entblockungsalgorithmus ATTDDBA in einem Anschlussfunktionsverwaltungsheader für LUTYPE6.1.

Für die Kommunikation zwischen zwei CICS-Systemen hat das Feld für den Entblockungsalgorithmus in einem Anschlussfunktionsverwaltungsheader für CICS keine besondere Bedeutung. Für die meisten CICS-Anwendungen kann die Option weggelassen werden.

Der “datenwert” ist ein Halbwort-Binärwert. Nur die 8 niedrigstwertigen Byte werden verwendet. Die SNA-definierten Bedeutungen der Bits sind folgende:

```
0-7   Reserviert - müssen auf null gesetzt werden.  
8-15  X'00' - Reserviert.  
      X'01' - Variable Länge.  
           Variabel geblockt.  
      X'02' - Reserviert.  
      X'03' - Reserviert.  
      X'04' - Kette von RUs (Anforderung/Antwort-Einheiten).  
      X'05' bis X'FF' - Reserviert.
```

Wenn die Option nicht angegeben wird, wird der Wert “Kette von RUs” angenommen.

RESOURCE(*name*)

Diese Option entspricht dem Ressourcennamen ATTPRN in einem Anschlussfunktionsverwaltungsheader für LUTYPE6.1.

RPROCESS(*name*)

Diese Option entspricht dem Rückgabeprozessnamen ATTRDPN in einem Anschlussfunktionsverwaltungsheader für LUTYPE6.1.

Für die Kommunikation zwischen zwei CICS-Systemen hat der Rückgabeprozessname in einem Anschlussfunktionsverwaltungsheader für CICS keine Bedeutung.

Informationen zur Kommunikation zwischen einem CICS-System und einem anderen Subsystem finden Sie in der für das Subsystem bereitgestellten Dokumentation zur Verwendung des Felds für den Rückgabeprozessnamen in einem Anschlussfunktionsverwaltungsheader (FMH).

RRESOURCE(*name*)

Diese Option entspricht dem Rückgaberessourcennamen ATTPRN in einem Anschlussfunktionsverwaltungsheader für LUTYPE6.1.

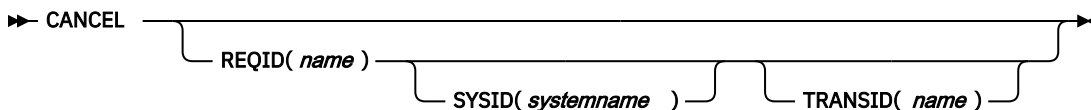
Für die Kommunikation zwischen zwei CICS-Systemen hat der Rückgaberessourcenname in einem Anschlussfunktionsverwaltungsheader für CICS keine Bedeutung.

Informationen zur Kommunikation zwischen einem CICS-System und einem anderen Subsystem finden Sie in der für das Subsystem bereitgestellten Dokumentation zur Verwendung des Felds für den Rückgaberessourcenname in einem Anschlussfunktionsverwaltungsheader (FMH).

CANCEL

Bricht Anforderungen zur Intervallsteuerung ab.

CANCEL



Bedingungen: ISCINVREQ, NOTAUTH, NOTFND, SYSIDERR

Hinweis für dynamisches Transaktionsrouting: Die Verwendung von CANCEL mit der Option REQID (eines Befehls POST, DELAY oder START) kann möglicherweise transaktionsübergreifende Affinitäten erstellen, die sich negativ auf die Verwendung des dynamischen Transaktionsroutings auswirken. Weitere Informationen zu Transaktionsaffinitäten finden Sie unter [Affinität](#)

Beschreibung

Der Befehl CANCEL bricht einen zuvor abgesetzten Befehl DELAY, POST oder START ab. Der Befehl CANCEL kann nicht dazu verwendet werden, eine Anforderung zu entfernen, die sich in einer lokalen Warteschlange befindet. Wenn Sie die Option SYSID angeben, wird der Befehl an ein fernes System übertragen. Geben Sie die Option SYSID nicht an, gibt die Option TRANSID an, wenn sie vorhanden ist, wo der Befehl ausgeführt werden soll. Der Effekt des Abbruchs unterscheidet sich je nach Befehlstyp, der abgebrochen wird, wie folgt:

- Ein Befehl DELAY kann nur abgebrochen werden, bevor er abläuft, und dies nur durch eine andere Task als die Task, die den Befehl DELAY abgesetzt hat (der für die Dauer der Anforderung ausgesetzt wird). Die Anforderungs-ID (REQID), die von der ausgesetzten Task verwendet wird, muss angegeben werden. Der Effekt des Abbruchs ist derselbe wie der bei einem frühen Ablauf des ursprünglichen Befehls DELAY. Das heißt, die ausgesetzte Task wird ebenso zuteilbar wie bei Erreichen der ursprünglichen Ablaufzeit.
- Wenn ein Befehl POST, der durch dieselbe Task abgesetzt wurde, abgebrochen werden soll, muss die Option REQID nicht angegeben werden. Der Abbruch kann entweder vor oder nach Ablauf der ursprünglichen Anforderung angefordert werden. Der Effekt des Abbruchs ist der gleiche, wie wenn die ursprüngliche Anforderung nie stattgefunden hätte.
- Wenn ein Befehl POST, der durch von einer anderen Task abgesetzt wurde, abgebrochen werden soll, muss in der Option REQID die Anforderungs-ID dieses Befehls angegeben werden. Der Effekt des Abbruchs ist derselbe wie der bei einem frühen Ablauf des ursprünglichen Befehls POST. Das heißt, der Steuerbereich für das Zeitgeberereignis für die andere Task wird so gepostet, als wäre die ursprüngliche Ablaufzeit erreicht worden.
- Wenn ein Befehl START abgebrochen werden soll, muss die Anforderungs-ID (REQID) des ursprünglichen Befehls angegeben werden. Der Effekt des Abbruchs ist der gleiche, wie wenn der ursprüngliche Befehl nie abgesetzt worden wäre. Der Abbruch ist nur wirksam, bevor der ursprüngliche Befehl berücksichtigt wird.
- Wenn Sie einen Befehl START mit der Option PROTECT verwenden, bricht der Befehl CANCEL den Befehl START nur ab, wenn der Befehl START festgeschrieben ist.

Anmerkung: Eine Antwort NOTFND auf einen Befehl CANCEL für einen Befehl START mit der Option REQID bedeutet, dass die Startanforderung nicht mehr aussteht. Sie impliziert nicht, dass die zu startende Task bis zu diesem Zeitpunkt abgeschlossen wurde. Sie impliziert auch nicht, dass die gestartete Task einen Befehl RETRIEVE abgesetzt hat, um die Daten in FROM aus der Warteschlange mit der REQID zu lesen. Ein nachfolgender Befehl START, der denselben Wert für REQID wiederverwendet, kann mit einem Abbruchcode AEIQ (Bedingung IOERR) fehlschlagen, wenn die Warteschlange mit der REQID zu diesem Zeitpunkt noch vorhanden ist.

Optionen

REQID(name)

Diese Option gibt einen Namen (1-8 Zeichen) an, der eindeutig sein muss, um den Befehl zu identifizieren. Dieser Name wird als temporäre Speicher-ID verwendet. Die temporäre Speicherwarteschlange, die auf diese Weise identifiziert wird, muss als lokale Warteschlange auf dem CICS-System definiert sein, auf dem der Befehl CANCEL verarbeitet wird.

Diese Option kann nicht zum Abbrechen eines Befehls POST verwendet werden, der durch dieselbe Task abgesetzt wurde (in diesem Fall wird die Option REQID ignoriert, falls sie angegeben wird).

SYSID(systemname)

(Nur ferne Systeme) Diese Option gibt den Namen (1-4 Zeichen) des Systems für den Befehl CANCEL an.

TRANSID(name)

Diese Option gibt die symbolische ID (1-4 Zeichen) für eine Transaktion an, die bei keiner Angabe von SYSID verwendet wird, um zu ermitteln, wo der Befehl CANCEL ausgeführt werden soll. Wenn die Op-

tion TRANSID als fern (REMOTE) definiert ist, wird die CANCEL-Anforderung durch Funktionsverlagerung an das ferne System übertragen.

Bedingungen

54 ISCINVREQ

Diese Bedingung tritt auf, wenn das ferne System einen Fehler angibt, der keiner bekannten Bedingung entspricht.

Standardaktion: Die Task wird abnormal beendet.

70 NOTAUTH

Diese Bedingung tritt auf, wenn eine Ressourcensicherheitsprüfung für die angegebene TRANSID oder die TRANSID des Befehls START, der der Anforderungs-ID entspricht, fehlschlägt.

Standardaktion: Die Task wird abnormal beendet.

13 NOTFND

Diese Bedingung tritt auf, wenn die angegebene Anforderungs-ID keinem noch nicht abgelaufenen Intervallsteuerungsbefehl entspricht.

Standardaktion: Die Task wird abnormal beendet.

53 SYSIDERR

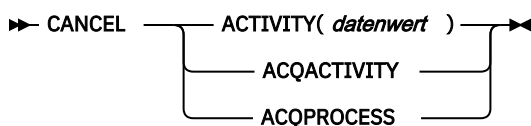
Diese Bedingung tritt auf, wenn die Option SYSID einen Namen angibt, der weder das lokale System noch ein fernes System (das in CICS durch die Definition einer Verbindung CONNECTION bekannt ist) bezeichnet. Sie tritt außerdem auf, wenn die Verbindung zu dem fernen System geschlossen ist.

Standardaktion: Die Task wird abnormal beendet.

CANCEL (BTS)

Bricht eine BTS-Aktivität oder einen BTS-Prozess ab.

CANCEL (BTS)



Bedingungen: ACTIVITYBUSY, ACTIVITYERR, INVREQ, IOERR, LOCKED, NOTAUTH, PROCESSBUSY, PROCESSERR

Beschreibung

Der Befehl CANCEL (BTS) erzwingt den Modus COMPLETE für eine BTS-Aktivität bzw. einen BTS-Prozess und alle zugehörigen untergeordneten Aktivitäten.

Optionen

ACQACTIVITY

Diese Option gibt an, dass diejenige Aktivität abgebrochen werden soll, die durch die aktuelle Arbeitseinheit mit einem Befehl ACQUIRE ACTIVITYID angefordert wurde.

ACQPROCESS

Diese Option gibt an, dass der Prozess abgebrochen werden soll, der durch die aktuelle Arbeitseinheit angefordert wurde.

ACTIVITY(datenwert)

Diese Option gibt die ID (1 bis 16 Zeichen) der abzubrechenden untergeordneten Aktivität an.

Bedingungen

107 ACTIVITYBUSY

RESP2-Werte:

19

Eine oder mehrere untergeordnete Aktivitäten der abzubrechenden Aktivität sind nicht zugänglich oder befinden sich im Modus CANCELLING.

109 ACTIVITYERR

RESP2-Werte:

8

Die in der Option ACTIVITY angegebene Aktivität wurde nicht gefunden.

14

Die abzubrechende Aktivität befindet sich nicht im Modus INITIAL oder DORMANT.

16 INVREQ

RESP2-Werte:

4

Mit der Option ACTIVITY wurde eine untergeordnete Aktivität angegeben, aber der Befehl wurde außerhalb des Geltungsbereichs einer gegenwärtig aktiven Aktivität ausgegeben.

15

Die Option ACQPROCESS wurde verwendet, aber die ausgebende Task hat keinen Prozess angefordert.

24

Die Option ACQACTIVITY wurde verwendet, die absetzende Task hat jedoch keine Aktivität angefordert.

17 IOERR

RESP2-Werte:

29

Die Repository-Datei ist nicht verfügbar.

30

Für die Repository-Datei trat ein Ein-/Ausgabefehler auf.

100 LOCKED

Die Anforderung kann nicht ausgeführt werden, weil eine Sperre für den relevanten Datensatz in der Repository-Datei besteht.

70 NOTAUTH

RESP2-Werte:

101

Der Benutzer, der der ausgebenden Task zugeordnet ist, ist nicht zum Zugriff auf die Datei berechtigt, die zur BTS-Repository-Datei gehört, in der Details des Prozesses bzw. der Aktivität gespeichert sind.

106 PROCESSBUSY

RESP2-Werte:

13

Eine oder mehrere der Aktivitäten, aus denen der abzubrechende Prozess besteht, sind nicht zugänglich oder befinden sich im Modus CANCELLING.

108 PROCESSERR

RESP2-Werte:

9

Der Prozesstyp wurde nicht gefunden.

14

Der abzubrechende Prozess befindet sich nicht im Modus INITIAL, DORMANT oder COMPLETE.

Aktivitäten

Ein Programm kann ausschließlich die folgenden Aktivitäten abbrechen:

- Seine eigenen untergeordneten Aktivitäten, falls es als Aktivierung einer Aktivität ausgeführt wird. Es kann mehrere seiner untergeordneten Aktivitäten innerhalb derselben Arbeitseinheit abbrechen.
- Die von ihm mit einem Befehl ACQUIRE ACTIVITYID angeforderte Aktivität in der aktuellen Arbeitseinheit.

Damit der Abbruch erfolgreich ausgeführt wird, muss sich eine Aktivität im Modus INITIAL oder DORMANT befinden. CICS versucht, Aktivitäten synchron abzuberechnen. Falls eine oder mehrere untergeordnete Aktivitäten der abzuberechnenden Aktivität jedoch nicht zugänglich sind (beispielsweise aufgrund eines Fehlers bei der Datenübertragungsverbindung), gilt Folgendes:

- Die untergeordnete Baumstruktur der untergeordneten Aktivitäten wird asynchron abgebrochen.
- Die abzuberechnende Aktivität wird in den Modus CANCELLING versetzt.

Das Beendigungsereignis, das einer abgebrochenen Aktivität zugeordnet ist, wird nicht aus dem Ereignispool des übergeordneten Elements gelöscht. Bei einer normalen Beendigung dieses Befehls ist die Aktivität weiterhin vorhanden und kann bei Bedarf zurückgesetzt sowie erneut ausgeführt werden.

Wenn eine angeforderte Aktivität abgebrochen wird, wird ihr übergeordnetes Element erneut aktiviert, weil das Beendigungsereignis für die abgebrochene Aktivität ausgelöst wird.

Prozesse

Ein Programm kann ausschließlich denjenigen Prozess abbrechen, den es in der aktuellen Arbeitseinheit angefordert hat. Falls es diesen Abbruch durchführt, kann es innerhalb der aktuellen Arbeitseinheit keinen weiteren Prozess anfordern.

Damit der Abbruch erfolgreich ausgeführt wird, muss sich ein Prozess im Modus INITIAL, DORMANT oder COMPLETE befinden.

CICS versucht, den Prozess synchron abzuberechnen; hierbei wird dasselbe Verfahren wie für Aktivitäten angewendet.

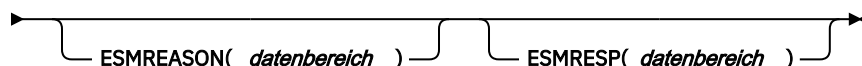
CHANGE PHRASE

Ändert das Kennwort oder die Kennwortphrase, das bzw. die von einem *externen Sicherheitsmanager (ESM)* für eine angegebene Benutzer-ID aufgezeichnet wurde.

CHANGE PHRASE

➡ CHANGEPHRASE(*datenbereich*) — PHRASELEN(*datenwert*) — NEWPHRASE(*datenbereich*) →

 ➡ NEWPHRASELEN(*datenwert*) — USERID(*datenwert*) →



Bedingungen: INVREQ, LENGERR, NOTAUTH, USERIDERR

Dieser Befehl ist threadsicher.

Beschreibung

Eine Benutzer-ID kann sowohl ein Kennwort als auch eine Kennwortphrase haben. Wenn der Wert von PHRASELEN zwischen 1 und 8 Zeichen liegt, wird die Phrase wie ein Kennwort behandelt. Liegt die Länge zwischen 9 und 100 Zeichen, wird der Wert wie eine Kennwortphrase behandelt. Sie können kein Kennwort mit einer Länge von 1 bis 8 Zeichen verwenden, um eine Kennwortphrase zu ändern. Ebenso wenig ist es möglich, mit einer Kennwortphrase aus 9 bis 100 Zeichen ein Kennwort mit einer Länge von 1 bis 8 Zeichen zu ändern.

Im Gegensatz zum Befehl **SIGNON** ist der Befehl **CHANGE PHRASE** nicht von der Hauptfunktion abhängig und kann dementsprechend auch in anderen als Terminalumgebungen, wie zum Beispiel in Webanwendungen und APPC-Sitzungen, abgesetzt werden.



Achtung: Um sicherzustellen, dass Kennwörter nicht in System- oder Transaktionsspeicherausfällen offengelegt werden, löschen Sie den Inhalt von Kennwort- bzw. Kennwortphrasenfeldern in den EXEC CICS-Befehlen, die eine Option für Kennwort oder Kennwortphrase haben, möglichst bald nach der Verwendung.

Optionen

Die Optionen ESMRESP und ESMREASON geben die Antwort- und Ursachencodes, sofern zutreffend, vom externen Sicherheitsmanager (ESM) zurück.

ESMREASON(datenbereich)

Gibt den Ursachencode in einem Vollwort-Binärfeld zurück, den CICS vom ESM empfängt.

Wenn es sich bei dem ESM um RACF handelt, enthält dieses Feld den RACF-Ursachencode.

ESMRESP(datenbereich)

Gibt den Antwortcode in einem Vollwort-Binärfeld zurück, das CICS vom ESM empfängt.

Wenn es sich bei dem ESM um RACF handelt, enthält dieses Feld den RACF-Rückgabecode.

NEWPHRASE(datenbereich)

Gibt ein optionales, 1 bis 8 Zeichen langes neues Kennwort oder eine neue, 9 bis 100 Zeichen lange Kennwortphrase an, die für den ESM erforderlich sind. Das Kennwort wird nur geändert, wenn das aktuelle Kennwort korrekt angegeben wird. Die Kennwortphrase wird nur geändert, wenn die aktuelle Kennwortphrase korrekt angegeben wird.

Falls der externe Sicherheitsmanager Kennwörter in Groß-/Kleinschreibung nicht zulässt, wird das 1 bis 8 Zeichen lange Kennwort in Großbuchstaben konvertiert.

NEWPHRASELEN(datenbereich)

Gibt die Länge des neuen Kennworts bzw. der neuen Kennwortphrase als Vollwort-Binärwert an.

PHRASE(datenbereich)

Gibt das aktuelle Kennwort oder die aktuelle Kennwortphrase der angegebenen Benutzer-ID an.

Falls der externe Sicherheitsmanager Kennwörter in Groß-/Kleinschreibung nicht zulässt, wird das 1 bis 8 Zeichen lange Kennwort in Großbuchstaben konvertiert.

PHRASELEN(datenbereich)

Gibt die Länge des aktuellen Kennworts bzw. der aktuellen Kennwortphrase als Vollwort-Binärwert an.

USERID(datenwert)

Gibt die Benutzer-ID des Benutzers an, dessen Kennwort oder Kennwortphrase geändert werden soll.

Die angegebene Benutzer-ID wird in Großbuchstaben konvertiert.

Bedingungen

16 INVREQ

RESP2-Werte:

2

Sie können kein Kennwort zum Ändern einer Kennwortphrase und keine Kennwortphrase zum Ändern eines Kennworts verwenden.

13

Der vom externen Sicherheitsmanager zurückgegebene Wert in der Option ESMRESP ist nicht von CICS klassifiziert. Eine Erläuterung der Optionen ESMRESP und ESMREASON finden Sie in der ESM-Dokumentation.

18

Die CICS-ESM-Schnittstelle ist nicht initialisiert.

29

Der ESM antwortet nicht.

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

RESP2-Werte:

1

Der Wert für PHRASELEN liegt außerhalb des gültigen Bereichs.

2

Der Wert für NEWPHRASELEN liegt außerhalb des gültigen Bereichs.

70 NOTAUTH

RESP2-Werte:

2

Das angegebene Kennwort bzw. die angegebene Kennwortphrase ist falsch. Wenn es sich bei dem ESM um RACF handelt, wird der von RACF verwaltete Zähler für Kennwortentziehungen inkrementiert.

4

Das neue Kennwort bzw. die neue Kennwortphrase ist nicht zulässig.

19

Die Benutzer-ID wurde entzogen.

20

Die Verbindung zur Standardgruppe des Benutzers wurde entzogen.

22

Die Kennwortänderungsanforderung ist während der SECLABEL-Verarbeitung fehlgeschlagen.

31

Der Benutzer wurde in der Verbindung zur Standardgruppe entzogen.

Standardaktion: Die Task wird abnormal beendet.

69 USERIDERR

RESP2-Werte:

8

Die Benutzer-ID (USERID) ist im ESM nicht bekannt.

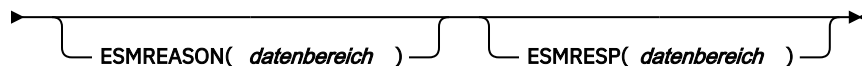
Standardaktion: Die Task wird abnormal beendet.

CHANGE PASSWORD

Ändert das Kennwort, das von einem *externen Sicherheitsmanager (ESM)* für eine angegebene Benutzer-ID aufgezeichnet wurde.

CHANGE PASSWORD

➡ CHANGEPASSWORD(*datenwert*) — NEWPASSWORD(*datenwert*) — USERID(*datenwert*) →



Bedingungen: INVREQ, NOTAUTH, USERIDERR

Dieser Befehl ist threadsicher.

Beschreibung

Im Gegensatz zum Befehl **SIGNON** ist der Befehl **CHANGE PASSWORD** nicht von der Hauptfunktion abhängig und kann dementsprechend auch in anderen als Terminalumgebungen, wie zum Beispiel in Webanwendungen und APPC-Sitzungen, abgesetzt werden.



Achtung: Sie sollten den Inhalt der Kennwortfelder in EXEC CICS-Befehlen, die eine Option für Kennwort haben, möglichst bald nach der Verwendung löschen. Dies stellt sicher, dass keine Kennwörter in System- oder Transaktionsspeicherauszügen offengelegt werden.

Optionen

Die Optionen ESMRESP und ESMREASON geben die Antwort- und Ursachencodes, sofern zutreffend, vom externen Sicherheitsmanager (ESM) zurück.

ESMREASON(*datenbereich*)

Gibt den Ursachencode in einem Vollwort-Binärfeld zurück, den CICS vom ESM empfängt.

Falls RACF als externer Sicherheitsmanager verwendet wird, enthält dieses Feld den RACF-Ursachencode.

ESMRESP(*datenbereich*)

Gibt den Antwortcode in einem Vollwort-Binärfeld zurück, das CICS vom ESM empfängt.

Falls RACF als externer Sicherheitsmanager verwendet wird, enthält dieses Feld den RACF-Antwortcode.

NEWPASSWORD(*datenwert*)

Gibt das neue, acht Zeichen lange Kennwort für die angegebene Benutzer-ID an. Das Kennwort wird nur geändert, wenn das aktuelle Kennwort korrekt angegeben wird.

Falls der externe Sicherheitsmanager Kennwörter in Groß-/Kleinschreibung nicht zulässt, wird das Kennwort in Großbuchstaben konvertiert.

PASSWORD(*datenwert*)

Gibt das aktuelle, acht Zeichen lange Kennwort für die angegebene Benutzer-ID an.

Falls der externe Sicherheitsmanager Kennwörter in Groß-/Kleinschreibung nicht zulässt, wird das Kennwort in Großbuchstaben konvertiert.

USERID(*datenwert*)

Gibt die Benutzer-ID des Benutzers an, dessen Kennwort geändert werden soll.

Die angegebene Benutzer-ID wird in Großbuchstaben konvertiert.

Bedingungen

16 INVREQ

RESP2-Werte:

13

Der vom externen Sicherheitsmanager zurückgegebene Wert in der Option ESMRESP ist nicht von CICS klassifiziert. Eine Erläuterung der Optionen ESMRESP und ESMREASON finden Sie in der ESM-Dokumentation.

18

Die CICS-ESM-Schnittstelle ist nicht initialisiert.

29

Der ESM antwortet nicht.

Standardaktion: Die Task wird abnormal beendet.

70 NOTAUTH

RESP2-Werte:

2

Das angegebene Kennwort ist falsch. Wenn es sich bei dem ESM um RACF handelt, wird der von RACF verwaltete Zähler für Kennwortentziehungen inkrementiert.

4

Das neue Kennwort ist nicht zulässig.

19

Die Benutzer-ID (USERID) wurde entzogen.

22

Die Kennwortänderungsanforderung ist während der SECLABEL-Verarbeitung fehlgeschlagen.

31

Der Benutzer wurde in der Verbindung zur Standardgruppe entzogen.

Standardaktion: Die Task wird abnormal beendet.

69 USERIDERR

RESP2-Werte:

8

Die Benutzer-ID (USERID) ist im ESM nicht bekannt.

Standardaktion: Die Task wird abnormal beendet.

CHANGE TASK

Ändert die Priorität einer Task.

CHANGE TASK

➡ CHANGE TASK ————— ➡
 └── **PRIORITY(*datenwert*)** ─┘

Bedingung: INVREQ

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl CHANGE TASK ändert die Priorität der Task, die den Befehl absetzt. Er hat sofortige Wirkung (im Gegensatz zum Befehl SET TASK), da die Steuerung während der Ausführung des Befehls abgegeben wird, sodass die aktuelle Task erneut zugeteilt werden muss. Die erneute Zuteilung findet erst statt, wenn Tasks, die eine höhere oder gleiche Priorität haben und die ebenfalls zur Ausführung bereit sind, zugeteilt wurden.

Wenn Sie die Option PRIORITY nicht angeben, verliert die Task die Steuerung nicht und die Priorität bleibt unverändert. Dies ist effektiv eine Nulloperation.

Optionen

PRIORITY(*datenwert*)

Die Option gibt einen Vollwort-Binärwert aus dem Bereich von 0-255 an, der die Priorität der Task definiert. Sie können auch den Wert -1 angeben, jedoch wird dadurch die Priorität weder geändert, noch die Task erneut zugeteilt.

Bedingungen

16 INVREQ

RESP2-Werte:

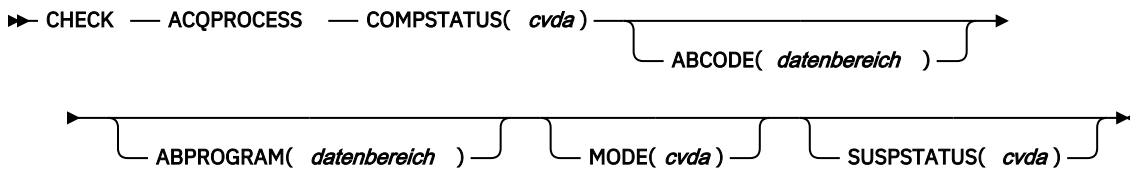
1

Der von Ihnen angegebene Wert für PRIORITY liegt außerhalb des Bereichs von -1 bis 255.

CHECK ACQPROCESS

Prüft den Beendigungsstatus eines BTS-Prozesses.

CHECK ACQPROCESS



Bedingung: INVREQ

Beschreibung

Der Befehl CHECK ACQPROCESS gibt den Beendigungsstatus des momentan abgerufenen BTS-Prozesses zurück. In der Regel wird der Befehl zum Prüfen des Erfolgs eines vorherigen Befehls RUN ACQPROCESS oder LINK ACQPROCESS verwendet. Er ermöglicht es dem Anforderer (Requester), zu erkennen, ob der Prozess erfolgreich abgeschlossen wurde oder ob er zum Beispiel erneut aktiviert werden muss, um die Verarbeitung abzuschließen.

Der einzige Prozess, den ein Programm prüfen kann, ist der Prozess, den es in der aktuellen Arbeitseinheit anfordert (siehe [Prozesse und Aktivitäten anfordern](#)).

Die Optionen RESP und RESP2 in diesem Befehl weisen darauf hin, ob der Befehl von CICS verstanden wird. Zum Beispiel tritt die Bedingung PROCESSERR auf, wenn der Prozess momentan vom Requester nicht angefordert ist.

Die Option COMPSTATUS gibt einen CVDA-Wert zurück, der den Beendigungsstatus der Stammaktivität des Prozesses angibt. Zum Beispiel wird der Status NORMAL zurückgegeben, wenn die Stammaktivität alle ihre Verarbeitungsschritte erfolgreich abschließt. Der Status INCOMPLETE wird zurückgegeben, wenn der Prozess von einer Aktivierung zurückkehrt, jedoch erneut aktiviert (angeschlossen) werden muss, um seine Verarbeitung abzuschließen.

Optionen

ABCODE(datenbereich)

Diese Option gibt den vier Zeichen langen Abbruchcode zurück, wenn die Stammaktivität des Prozesses abnormal beendet wurde.

ABPROGRAM(datenbereich)

Diese Option gibt den acht Zeichen langen Namen des Programms zurück, das die Steuerung zum Zeitpunkt der abnormalen Beendigung innehatte, wenn die Stammaktivität des Prozesses abnormal beendet wurde.

ACQPROCESS

Diese Option gibt an, dass der Prozess, der zurzeit durch den Requester angefordert ist, geprüft werden soll.

COMPSTATUS(cvda)

Diese Option gibt den Beendigungsstatus des Prozesses an. Die CVDA-Werte lauten wie folgt:

ABEND

Das Programm, das die Stammaktivität des Prozesses implementiert, wurde abnormal beendet. Alle untergeordneten Aktivitäten der Stammaktivität wurden abgebrochen.

FORCED

Das Programm wurde zur Beendigung gezwungen, zum Beispiel indem es durch einen Befehl CANCEL ACQPROCESS abgebrochen wurde.

INCOMPLETE

Der Prozess ist unvollständig. Dies kann Folgendes bedeuten:

- Er wurde noch nicht ausgeführt.
- Er ist von einer oder mehreren Aktivierungen zurückgekehrt, muss jedoch erneut aktiviert (angeschlossen) werden, um alle seine Verarbeitungsschritte abzuschließen.

- Er ist zurzeit aktiv.

NORMAL

Der Prozess wurde erfolgreich abgeschlossen.

MODE(cvda)

Diese Option gibt den Verarbeitungsstatus des Prozesses an. Die CVDA-Werte lauten wie folgt:

ACTIVE

Eine Aktivierung des Prozesses wird gerade ausgeführt.

CANCELLING

CICS wartet darauf, den Prozess abubrechen. Es wurde ein Befehl CANCEL ACQPROCESS abgesetzt, jedoch kann CICS den Prozess nicht sofort abbrechen, weil eine oder mehrere der untergeordneten Aktivitäten der Stammaktivität nicht zugänglich sind.

COMPLETE

Der Prozess ist abgeschlossen.

DORMANT

Der Prozess wartet darauf, dass ein Ereignis seine nächste Aktivierung auslöst.

INITIAL

Es wurde noch kein Befehl RUN oder LINK für den Prozess abgesetzt.

SUSPSTATUS(cvda)

Diese Option gibt an, ob der Prozess zurzeit ausgesetzt ist. Die CVDA-Werte lauten wie folgt:

SUSPENDED

Der Prozess ist ausgesetzt. Wenn ein Reaktivierungsereignis (erneuter Anschluss) auftritt, wird er nicht reaktiviert.

NOTSUSPENDED

Der Prozess ist zurzeit nicht ausgesetzt. Wenn ein Reaktivierungsereignis auftritt, wird er reaktiviert.

Bedingungen

16 INVREQ

RESP2-Werte:

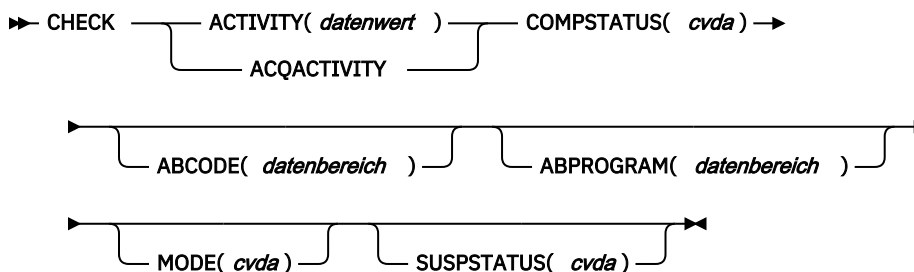
15

Die Arbeitseinheit, die die Anforderung absetzt, hat keinen Prozess angefordert.

CHECK ACTIVITY

Prüft den Beendigungsstatus einer BTS-Aktivität.

CHECK ACTIVITY



Bedingungen: ACTIVITYBUSY, ACTIVITYERR, INVREQ, IOERR, LOCKED

Beschreibung

Der Befehl CHECK ACTIVITY gibt den Beendigungsstatus einer BTS-Aktivität zurück. In der Regel wird der Befehl zum Prüfen des Erfolgs eines vorherigen Befehls RUN ACTIVITY oder LINK ACTIVITY verwendet. Er ermöglicht es dem Anforderer (Requester), zu erkennen, ob die Aktivität erfolgreich abgeschlossen wurde oder ob sie zum Beispiel erneut aktiviert werden muss, um die Verarbeitung abzuschließen.

Der Befehl CHECK ACTIVITY kann abgesetzt werden:

1. Von einer übergeordneten Aktivität, um den Beendigungsstatus einer der zugehörigen untergeordneten Aktivitäten zu prüfen.
2. Von einem Programm, das eine Aktivität mit dem Befehl ACQUIRE ACTIVITYID angefordert hat.

Der Befehl kann zum Prüfen der folgenden untergeordneten Aktivitäten (nicht Stammaktivitäten) verwendet werden:

- Aktivitäten, die abgeschlossen wurden.
- Aktivitäten, die nicht abgeschlossen wurden.
- Aktivitäten, die zur asynchronen Ausführung angefordert wurden.
- Aktivitäten, die zur synchronen Ausführung angefordert wurden.

Die Optionen RESP und RESP2 weisen darauf hin, ob der Befehl von CICS— verstanden wird. Zum Beispiel tritt die Bedingung ACTIVITYERR auf, wenn die in der Option ACTIVITY angegebene untergeordnete Aktivität für die übergeordnete Aktivität nicht definiert ist.

Die Option COMPSTATUS gibt einen CVDA-Wert zurück, der den Beendigungsstatus der Aktivität angibt. Zum Beispiel wird der Status NORMAL zurückgegeben, wenn die Aktivität alle ihre Verarbeitungsschritte erfolgreich abschließt. Der Status INCOMPLETE wird zurückgegeben, wenn sie von einer Aktivierung zurückkehrt, jedoch erneut aktiviert (angeschlossen) werden muss, um ihre Verarbeitung abzuschließen.

Wenn dieser Befehl von einer übergeordneten Aktivität im Hinblick auf eine ihrer untergeordneten Aktivitäten abgesetzt wurde und die untergeordnete Aktivität abgeschlossen ist, löscht CICS nach der Rückkehr des Befehls das Beendigungsereignis der untergeordneten Aktivität aus dem Ereignispool der übergeordneten Aktivität.

Weitere Informationen zur Verwendung des Befehls CHECK ACTIVITY finden Sie unter [Umgang mit BTS-Fehlern und -Antwortcodes](#).

Optionen

ABCODE(datenbereich)

Diese Option gibt den vier Zeichen langen Abbruchcode zurück, wenn die Aktivität abnormal beendet wurde.

ABPROGRAM(datenbereich)

Diese Option gibt den acht Zeichen langen Namen des Programms zurück, das die Steuerung zum Zeitpunkt der abnormalen Beendigung innehatte, wenn die Aktivität abnormal beendet wurde.

ACQACTIVITY

Diese Option gibt an, dass die zu prüfende Aktivität diejenige ist, die von der aktuellen Arbeitseinheit durch einen Befehl ACQUIRE ACTIVITYID angefordert wurde.

ACTIVITY(datenwert)

Diese Option gibt den Namen (1-16 Zeichen) der Aktivität an, die geprüft werden soll.

Verwenden Sie diese Option, um den Status einer untergeordneten Aktivität der aktuellen Aktivität zu prüfen.

COMPSTATUS(cvda)

Diese Option gibt den Beendigungsstatus der Aktivität an. Die CVDA-Werte lauten wie folgt:

ABEND

Das Programm, das die Aktivität implementiert, wurde abnormal beendet. Alle untergeordneten Aktivitäten der Aktivität wurden abgebrochen.

Das Beendigungsereignis der Aktivität wird aus dem Ereignispool der übergeordneten Aktivität gelöscht.

FORCED

Die Aktivität wurde zur Beendigung gezwungen, zum Beispiel indem sie durch einen Befehl CANCEL ACTIVITY abgebrochen wurde.

Das Beendigungsereignis der Aktivität wird aus dem Ereignispool der übergeordneten Aktivität gelöscht.

INCOMPLETE

Die angegebene Aktivität ist unvollständig. Dies kann Folgendes bedeuten:

- Sie wurde noch nicht ausgeführt.
- Sie ist von einer oder mehreren Aktivierungen zurückgekehrt, muss jedoch erneut aktiviert (angeschlossen) werden, um alle ihre Verarbeitungsschritte abzuschließen.
- Sie ist zurzeit aktiv.

Das Beendigungsereignis der Aktivität wird **nicht** aus dem Ereignispool der übergeordneten Aktivität gelöscht.

NORMAL

Die angegebene Aktivität wurde erfolgreich abgeschlossen.

Das Beendigungsereignis der Aktivität wird aus dem Ereignispool der übergeordneten Aktivität gelöscht.

MODE(cvda)

Diese Option gibt den Verarbeitungsstatus der Aktivität an. Die CVDA-Werte lauten wie folgt:

ACTIVE

Eine Aktivierung der Aktivität wird gerade ausgeführt.

CANCELLING

CICS wartet darauf, die Aktivität abubrechen. Es wurde ein Befehl CANCEL ACTIVITY abgesetzt, jedoch kann CICS die Aktivität nicht sofort abbrechen, weil eine oder mehrere der untergeordneten Aktivitäten der Aktivität nicht zugänglich sind.

COMPLETE

Die Aktivität ist abgeschlossen.

DORMANT

Die Aktivität wartet darauf, dass ein Ereignis ihre nächste Aktivierung auslöst.

INITIAL

Es wurde noch kein Befehl RUN oder LINK für die Aktivität abgesetzt oder die Aktivität wurde durch einen Befehl RESET ACTIVITY zurückgesetzt.

SUSPSTATUS(cvda)

Diese Option gibt an, ob die Aktivität zurzeit ausgesetzt ist. Die CVDA-Werte lauten wie folgt:

SUSPENDED

Die Aktivität ist ausgesetzt. Wenn ein Reaktivierungsereignis (erneuter Anschluss) auftritt, wird sie nicht reaktiviert.

NOTSUSPENDED

Die Aktivität ist zurzeit nicht ausgesetzt. Wenn ein Reaktivierungsereignis auftritt, wird sie reaktiviert.

Bedingungen

107 ACTIVITYBUSY

RESP2-Werte:

19

Die Anforderung hat das Zeitlimit überschritten. Es kann sein, dass die Beendigung einer anderen Task, die diesen Aktivitätsdatensatz verwendet, verhindert wurde.

109 ACTIVITYERR

RESP2-Werte:

8

Die in der Option ACTIVITY angegebene Aktivität konnte nicht gefunden werden.

16 INVREQ

RESP2-Werte:

4

Mit der Option ACTIVITY wurde eine untergeordnete Aktivität angegeben, aber der Befehl wurde außerhalb des Geltungsbereichs einer gegenwärtig aktiven Aktivität ausgegeben.

24

Die Option ACQACTIVITY wurde verwendet, jedoch hat die Arbeitseinheit, die die Anforderung abgesetzt hat, keine Aktivität angefordert.

17 IOERR

RESP2-Werte:

29

Die Repository-Datei ist nicht verfügbar.

30

Es ist ein Ein-/Ausgabefehler für die Repository-Datei aufgetreten.

100 LOCKED

Die Anforderung kann nicht ausgeführt werden, weil eine Sperre für den relevanten Datensatz in der Repository-Datei besteht.

CHECK TIMER

Prüft den Status eines BTS-Zeitgebers.

CHECK TIMER

➡ CHECK — TIMER(*datenwert*) — STATUS(*cvda*) ➡

Bedingungen: INVREQ, IOERR, TIMERERR

Beschreibung

Der Befehl CHECK TIMER gibt den Status eines BTS-Zeitgebers zurück. Mit diesem Befehl kann der Anforderer (Requester) ermitteln, ob ein Zeitgeber abgelaufen ist und, wenn dies der Fall ist, ob er normal abgelaufen ist oder ob der Ablauf durch einen Befehl FORCE TIMER erzwungen wurde.

Bei Rückkehr dieses Befehls wird das zugeordnete Ereignis, sofern der Zeitgeber abgelaufen ist, aus dem Ereignispool der aktuellen Aktivität gelöscht.

Die einzigen Zeitgeber, die ein Programm prüfen kann, sind diejenigen, deren Eigner die aktuelle Aktivität ist.

Optionen

STATUS(*cvda*)

Diese Option gibt den Status des Zeitgebers an. Die CVDA-Werte lauten wie folgt:

EXPIRED

Der Zeitgeber ist normal abgelaufen.

Das zugeordnete Ereignis wird aus dem Ereignispool der aktuellen Aktivität gelöscht.

FORCED

Der Zeitgeber ist abgelaufen, weil für ihn ein Befehl FORCE TIMER abgesetzt wurde.

Das zugeordnete Ereignis wird aus dem Ereignispool der aktuellen Aktivität gelöscht.

UNEXPIRED

Der Zeitgeber ist noch nicht abgelaufen.

Das zugeordnete Ereignis wird nicht aus dem Ereignispool der aktuellen Aktivität gelöscht.

TIMER(datenwert)

Diese Option gibt den Namen (1-16 Zeichen) des Zeitgebers an, der geprüft werden soll.

Bedingungen

16 INVREQ

RESP2-Werte:

1

Der Befehl wurde außerhalb des Gültigkeitsbereichs einer momentan aktiven Aktivität abgesetzt.

17 IOERR

Es ist ein Ein-/Ausgabefehler für die Repository-Datei aufgetreten.

115 TIMERERR

RESP2-Werte:

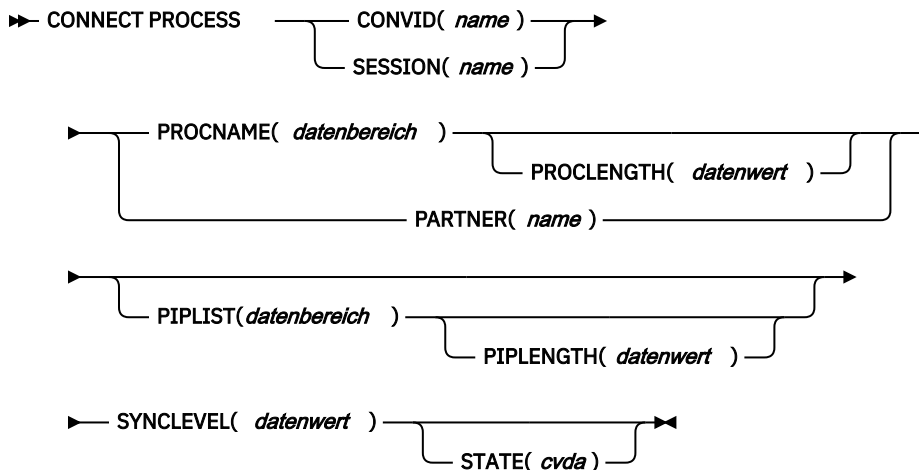
13

Der Zeitgeber, der in der Option TIMER angegeben wurde, ist nicht vorhanden.

CONNECT PROCESS

Initialisiert einen formatfreien APPC-Dialog (formatfreien APPC-Datenaustausch).

CONNECT PROCESS (APPC)



Bedingungen: INVREQ, LENGERR, NOTALLOC, PARTNERIDERR, TERMERR

Beschreibung

Mit dem Befehl CONNECT PROCESS kann eine Anwendung einen Prozessnamen und eine Synchronisationsebene angeben, die an CICS übergeben und verwendet werden sollen, wenn der ferne Partner angeschlossen wird.

Optionen

CONVID(name)

Diese Option gibt den Dialog an, auf den sich der Befehl bezieht. Der vier Zeichen lange Name gibt das Token an, das von einem zuvor ausgeführten Befehl ALLOCATE in EIBRSRCE im EIB zurückgegeben wurde.

Aus Gründen der Kompatibilität mit früheren Releases wird SESSION als Synonym für CONVID akzeptiert. Neue Programme sollten CONVID verwenden.

PARTNER(name)

Diese Option gibt den Namen (8 Zeichen) einer Gruppe von Definitionen an, die den Namen (oder erweiterten Namen) einer fernen Partnertransaktion (TPNAME oder XTPNAME) einschließt. Sie können diese Option als Alternative zu PROCNAME und PROCLength verwenden.

PIPLENGTH(datenwert)

Diese Option gibt die Gesamtlänge (Halbwort-Binärwert) der angegebenen Liste der Prozessinitialisierungsparameter (PIP) an.

PIPLIST(datenbereich)

Diese Option gibt die PIP-Daten an, die an das ferne System gesendet werden sollen. Die PIP-Liste besteht aus Datensätzen variabler Länge, die jeweils einen einzelnen Prozessinitialisierungsparameter (PIP) enthalten. Ein PIP beginnt mit einem zwei Byte langen inklusiven Längenfeld (LL), auf das ein zwei Byte langes reserviertes Feld und anschließend die Parameterdaten folgen.

PROCLength(datenwert)

Diese Option gibt die Länge des Namens, der durch die Option PROCNAME angegeben wird, als Halbwort-Binärwert aus dem Bereich von 1 bis 64 an.

PROCNAME(datenbereich)

Diese Option gibt den Partnerprozess (d. h. die Transaktion) an, die im fernen System angeschlossen werden soll.

Zur Identifikation einer CICS-Transaktion reicht ein Byte aus. Die APPC-Architektur lässt einen Bereich von 1 bis 64 Byte zu, jedoch darf jedes Produkt den eigenen Maximalwert selbst festlegen. CICS hält diese Vorgabe ein, indem ein Bereich von 1 bis 64 Byte zugelassen wird. Wenn das ferne System ein CICS-System ist, kann diese Option die vier Byte lange Transaktions-ID oder den Wert von TPNAME angeben, die bzw. der in der relevanten TRANSACTION-Definition angegeben ist. Alternativ können Sie die vollständige ID untersuchen, indem Sie den Benutzerexit XZCATT codieren.

Es wird keine Zeichenprüfung für den TPN von CICS durchgeführt.

Programmierungsinformationen zum Benutzerexit XZCATT finden Sie unter [Format von CICS-Statistikdatensätzen](#).

SESSION(name)

Diese Option gibt die symbolische ID (1-4 Zeichen) für einen Terminaleintrag in der Terminalsteuertabelle (TCTTE) für die Sitzung an. Diese Option gibt die alternative Funktion an, die verwendet werden soll.

STATE(cvda)

Diese Option ruft den Status des aktuellen Dialogs ab. Die folgenden CVDA-Werte können von CICS zurückgegeben werden:

- ALLOCATED
- CONFFREE
- CONFRECEIVE
- CONFSEND
- FREE
- PENDFREE
- PENDRECEIVE
- RECEIVE
- ROLLBACK
- SEND
- SYNCFREE
- SYNCRECEIVE
- SYNCSEND

SYNCLEVEL(datenwert)

Diese Option gibt die Synchronisationsebene (als Halbwort-Binärwert) für den aktuellen Dialog an. Die folgenden Werte sind möglich:

- 0 Keine
- 1 Bestätigen
- 2 Synchronisationspunkt

Bedingungen**16 INVREQ**

RESP2-Werte:

200

Eine Distributed Program Link-Serveranwendung hat die Funktionsverlagerungssitzung (ihre Hauptfunktion) in der Option CONVID angegeben.

Tritt außerdem in einer der folgenden Situationen auf (RESP2 nicht festgelegt):

- In der Option SYNCLEVEL wurde eine andere Synchronisationsebene als 0, 1 oder 2 angefordert.
- Der Befehl ist für das verwendete Terminal oder die verwendete logische Einheit (LU) nicht gültig.
- Der Befehl wird in einem Dialog verwendet, der zurzeit von CPI Communications verwendet wird oder bei dem es sich um einen APPC-Basisdialog handelt. Im letzteren Fall sollte der Befehl GDS CONNECT PROCESS verwendet werden.

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

Diese Bedingung tritt in einer der folgenden Situationen auf:

- In der Option PROCLength ein außerhalb des gültigen Bereichs liegender Wert angegeben.
- Der in der Option PIPLength angegebene Wert ist kleiner als 0.
- Der in der Option PIPLength angegebene Wert überschreitet die CICS-Implementierungsbegrenzung von 32 763.
- Ein Längenelement (LL) für PIPLIST hat einen Wert kleiner als 4.
- Die Summe der Längenelemente (LLs) in der PIP-Liste (PIPLIST) ist nicht gleich dem Wert, der durch PIPLength angegeben wird.

Standardaktion: Die Task wird abnormal beendet.

61 NOTALLOC

Diese Bedingung tritt auf, wenn sich der in CONVID angegebene Wert nicht auf einen Dialog bezieht, dessen Eigner die Anwendung ist.

Standardaktion: Die Task wird abnormal beendet.

97 PARTNERIDERR

Diese Bedingung tritt auf, wenn der Name, der in der Option PARTNER angegeben wurde, von CICS nicht erkannt wird.

Standardaktion: Die Task wird abnormal beendet.

81 TERMERR

Diese Bedingung tritt für einen sitzungsbezogenen Fehler auf. Jede Aktion außer einem Befehl FREE für diesen Dialog verursacht einen Abbruch mit dem Code ATCV.

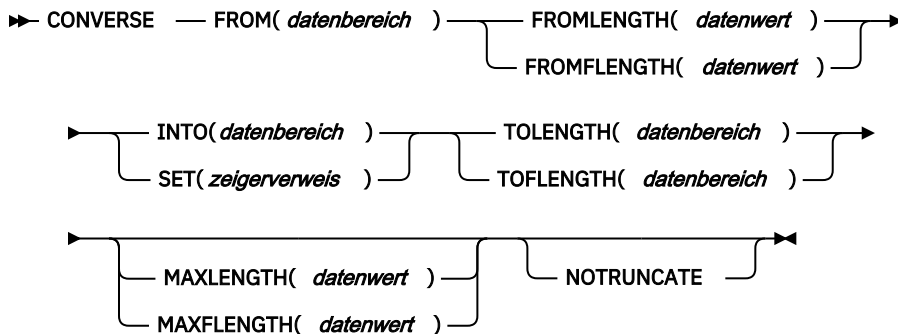
Eine Anforderung CANCEL TASK durch ein Benutzerknotenfehlerprogramm (NEP) kann diese Bedingung auslösen, wenn die Task eine ausstehende aktive Terminalsteueranforderung hat, wenn das Programm für abnormale Knotenbedingungen den Sitzungsfehler verarbeitet.

Standardaktion: Die Task wird abnormal mit dem Abbruchcode ATNI beendet.

CONVERSE (Standard)

Kommuniziert in der CICS-Standardterminalunterstützung.

CONVERSE (Standard)



Bedingungen: LENGERR

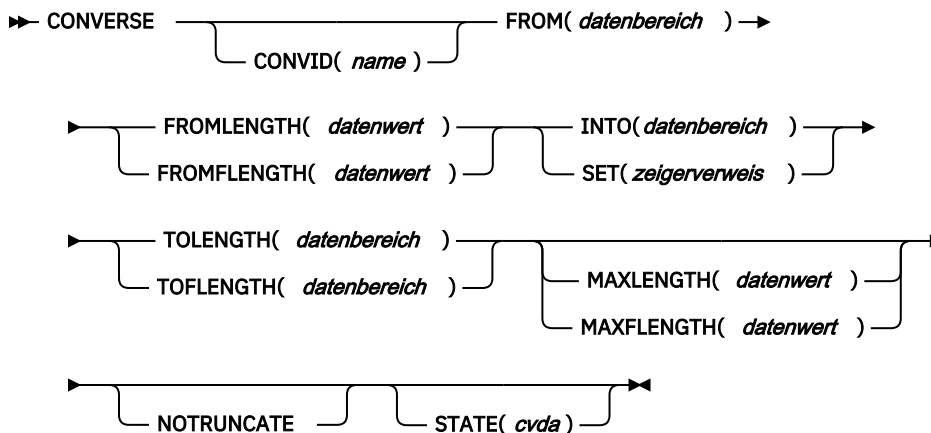
Beschreibung

Diese Form des Befehls CONVERSE wird von allen von CICS unterstützten z/OS Communications Server-Terminals verwendet, für die die anderen CONVERSE-Beschreibungen nicht geeignet sind.

CONVERSE (APPC)

Kommuniziert in einem formatfreien APPC-Dialog (formatfreien APPC-Datenaustausch).

CONVERSE (APPC)



Bedingungen: EOC, INVREQ, LENGERR, NOTALLOC, SIGNAL, TERMERR

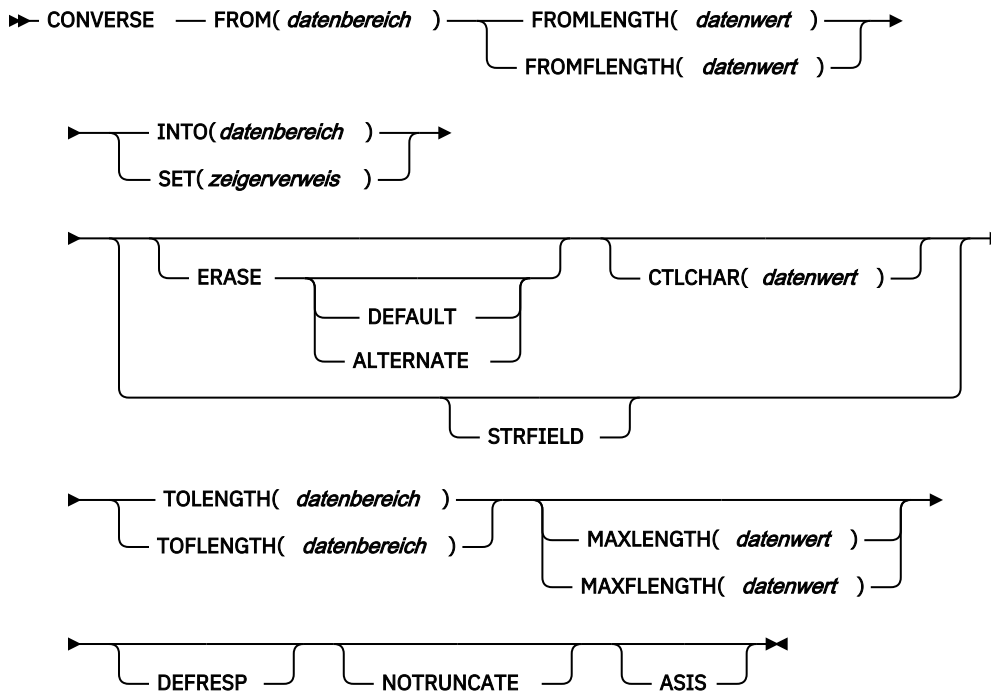
Beschreibung

Der Befehl CONVERSE sendet und empfängt anschließend Daten in einem formatfreien APPC-Dialog.

CONVERSE (LUTYPE2/LUTYPE3)

Kommuniziert auf einer logischen Einheit mit 3270-Anzeige (LUTYPE2) oder mit 3270-Drucker (LUTYPE3).

CONVERSE (LUTYPE2/LUTYPE3)



Bedingungen: EOC, LENGERR, TERMERR

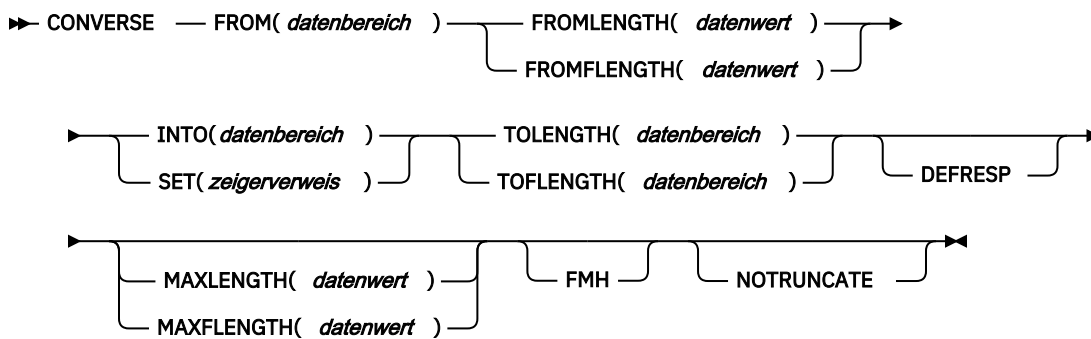
Beschreibung

Der Befehl CONVERSE kommuniziert auf einer logischen Einheit mit 3270-Anzeige oder 3270-Drucker.

CONVERSE (LUTYPE4)

Kommuniziert auf einer logischen Einheit mit LU-Typ 4 (LUTYPE4).

CONVERSE (LUTYPE4)



Bedingungen: EOC, EODS, IGREQCD, INBFMH, LENGERR, SIGNAL, TERMERR

Beschreibung

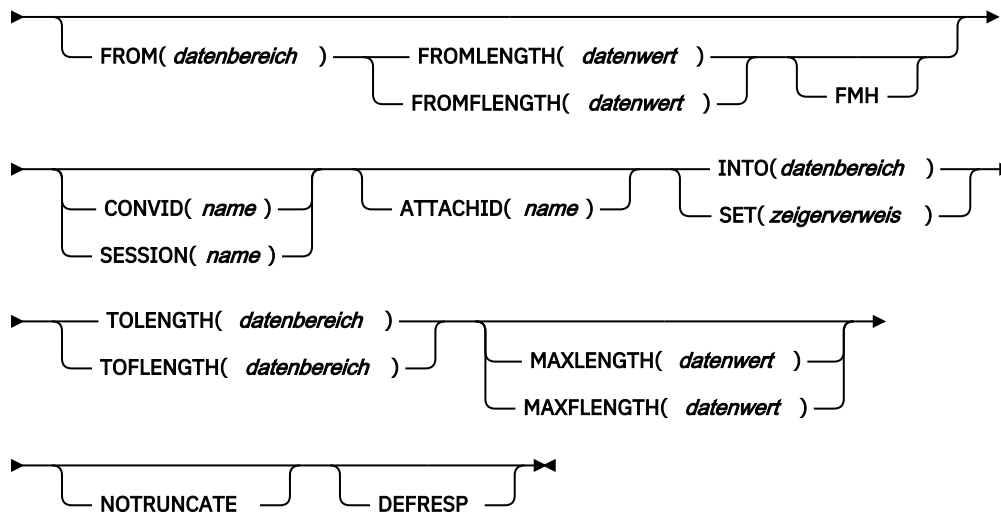
Der Befehl CONVERSE kommuniziert auf einer logischen Einheit mit LU-Typ 4 (LUTYPE4).

CONVERSE (LUTYPE6.1)

Kommuniziert auf einer logischen Einheit mit LU-Typ 6.1 (LUTYPE6.1).

CONVERSE (LUTYPE6.1)

➤ CONVERSE ➔



Bedingungen: CBIDERR, EOC, INBFMH, LENGERR, NOTALLOC, SIGNAL, TERMERR

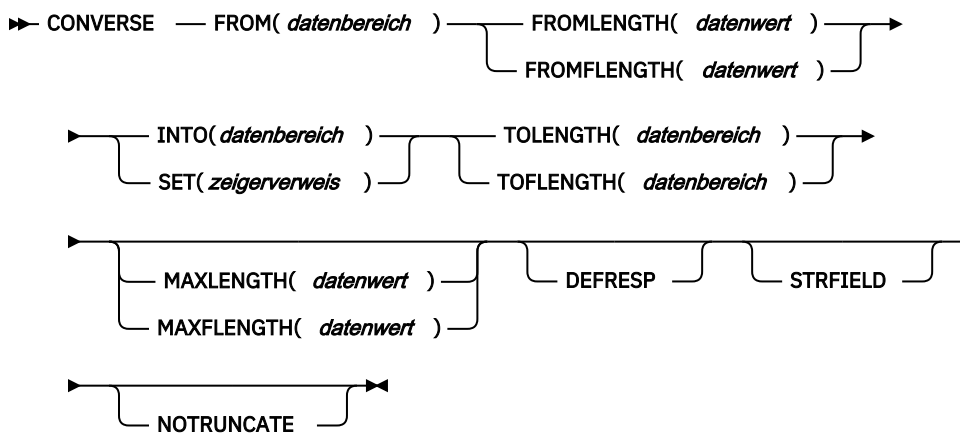
Beschreibung

Der Befehl CONVERSE kommuniziert auf einer logischen Einheit mit LU-Typ 6.1 (LUTYPE6.1).

CONVERSE (SCS)

Kommuniziert auf einer logischen Einheit mit 3270-SCS-Drucker.

CONVERSE (SCS)



Bedingungen: LENGERR, TERMERR

Beschreibung

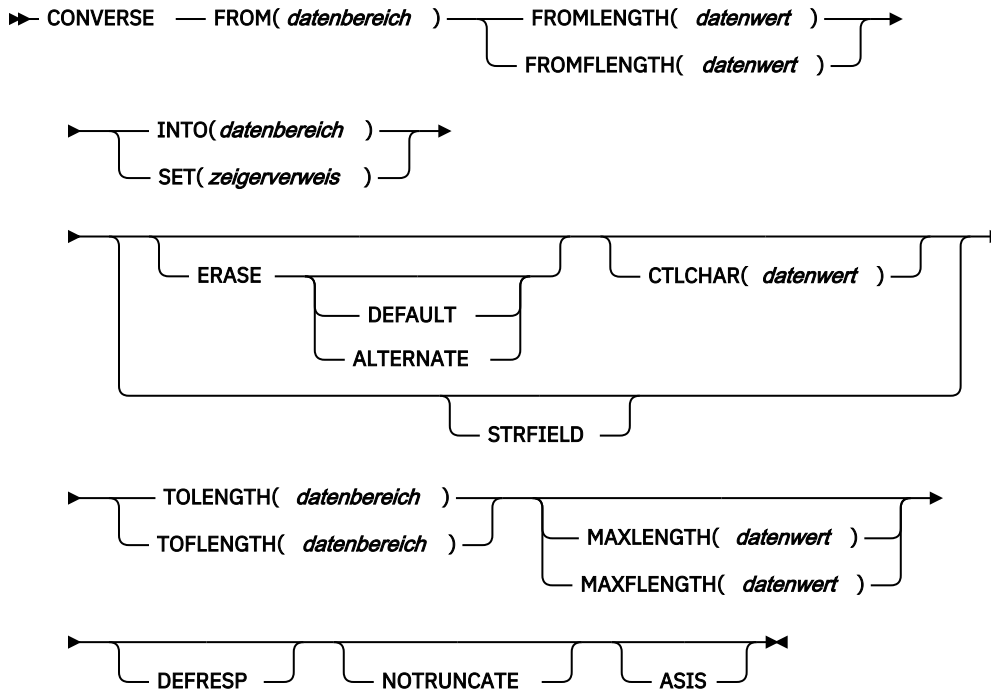
Der Befehl CONVERSE kommuniziert auf einer logischen Einheit 3270 mit SNA-Zeichenfolgedrucker (SCS-Drucker). Die logische SCS-Druckereinheit akzeptiert eine Zeichenfolge, wie sie durch die Systems Network Architecture (SNA) definiert wird. Einige Einheiten, die unter SNA verbunden sind, können ein Signal senden, das von dem Befehl HANDLE CONDITION SIGNAL erkannt wird, sodass er eine geeignete Behandlungsroutine aufrufen kann. Falls erforderlich, kann das Anwendungsprogramm mit einem Befehl

WAIT SIGNAL veranlasst werden, auf das Signal zu warten. Die Programmabrufstasten (PA) auf einer Einheit 3287 können in dieser Weise oder mit einem Befehl RECEIVE verwendet werden.

CONVERSE (3270 logisch)

Kommuniziert auf einer logischen Einheit 3270.

CONVERSE (3270 logisch)



Bedingungen: INVREQ, LENGERR, TERMERR

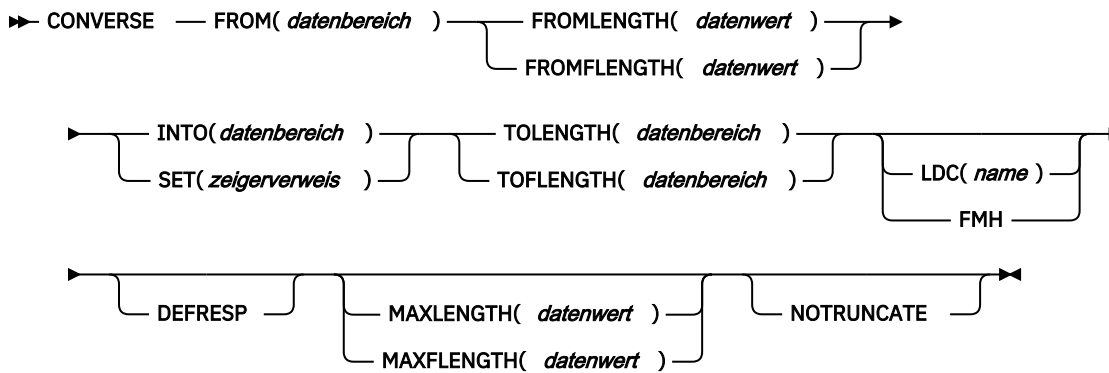
Beschreibung

Der Befehl CONVERSE kommuniziert auf einer logischen Einheit 3270.

CONVERSE (3600-3601)

Kommuniziert auf einer logischen Einheit 3600 (3601).

CONVERSE (3600-3601)



Bedingungen: EOC, EODS, INBFMH, LENGERR, SIGNAL, TERMERR

Beschreibung

Der Befehl CONVERSE kommuniziert auf einer logischen Einheit 3600. Diese Form des Befehls CONVERSE gilt auch für das Werkskommunikationssystem 4700 und 3630 (Plant Communication System).

Ein logischer Einheitencode (LDC) ist ein Code, der in einen abgehenden Funktionsverwaltungsheader (Function Management Header - FMH) eingeschlossen werden kann, um die Disposition der Daten anzugeben (z. B. an welches Subsystemterminal sie gesendet werden sollen). Jeder Code kann durch einen eindeutigen mnemonischen LDC-Code dargestellt werden.

Die Installation kann bis zu 256 zwei Zeichen lange mnemonische Codes für jeden TCTTE angeben und zwei oder mehr TCTTEs können eine Liste dieser mnemonischen Codes gemeinsam nutzen. Jedem mnemonischen LDC für jeden TCTTE entspricht jeweils ein numerischer Wert (0 - 255).

Einem LDC wird auch eine Einheit 3600 und eine logische Seitengröße zugeordnet. Mit LDC oder *LDC-Wert* wird in diesen Informationen der Code bezeichnet, der vom Benutzer angegeben wird. Die Bezeichnung *mnemonischer LDC* bezieht sich auf das zwei Zeichen lange Symbol, das den numerischen LDC-Wert darstellt.

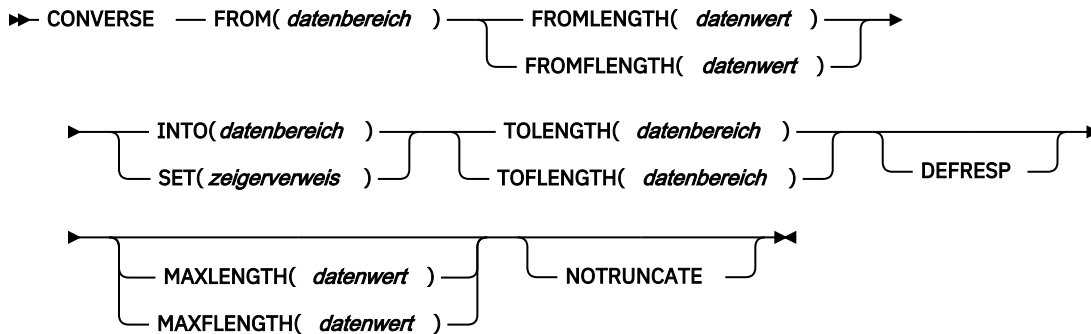
Wenn die Option LDC in dem Befehl CONVERSE angegeben wird, wird der numerische Wert, der dem mnemonischen Code für den bestimmten TCTTE zugeordnet ist, in den Funktionsverwaltungsheader (FMH) eingefügt. Dieser Wert wird durch die Installation ausgewählt und vom 3601-Anwendungsprogramm interpretiert.

Bei der Ausgabe kann der Funktionsverwaltungsheader (FMH) vom Anwendungsprogramm oder von CICS erstellt werden. Wenn Ihr Programm den Funktionsverwaltungsheader bereitstellt, können Sie ihn an den Anfang Ihrer Ausgabedaten stellen und die Option FMH in Ihrem Befehl CONVERSE angeben. Wenn Sie die Option FMH nicht angeben, stellt CICS einen FMH bereit, jedoch müssen Sie die ersten drei Byte der Nachricht zur Angabe durch CICS reservieren.

CONVERSE (3600-3614)

Kommuniziert auf einer logischen Einheit 3600 (3614).

CONVERSE (3600-3614)



Bedingungen: LENGERR, TERMERR

Beschreibung

Der Befehl CONVERSE kommuniziert auf einer logischen Einheit 3600.

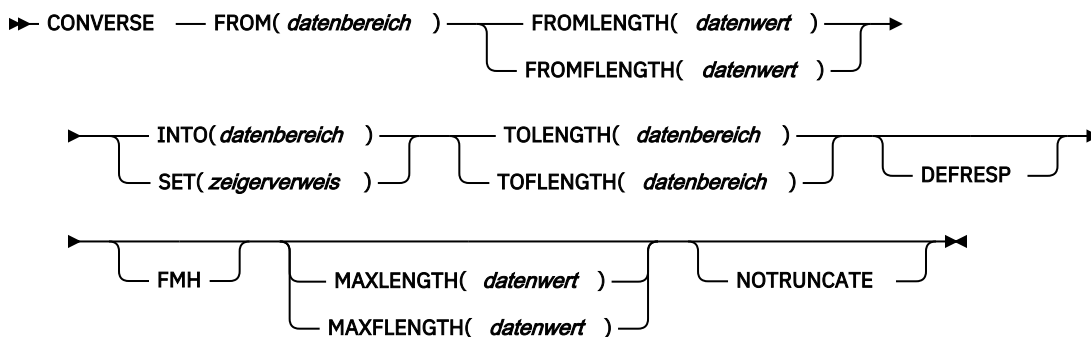
Das Datenstrom- und Kommunikationsformat, das zwischen einem CICS-Anwendungsprogramm und einer logischen Einheit 3614 verwendet wird, wird durch die logische Einheit 3614 bestimmt. Das Anwendungsprogramm ist dementsprechend einheitenabhängig, wenn es die 3614-Kommunikation verarbeitet.

Weitere Informationen zum Entwerfen von 3614-Anwendungsprogrammen für CICS finden Sie in der Veröffentlichung [IBM 4700/3600/3630 Guide](#).

CONVERSE (3650-Interpreter)

Kommuniziert auf einer logischen Einheit 3650 mit Interpreter.

CONVERSE (3650-Interpreter)



Bedingungen: EOC, EODS, INBFMH, LENGERR, TERMERR

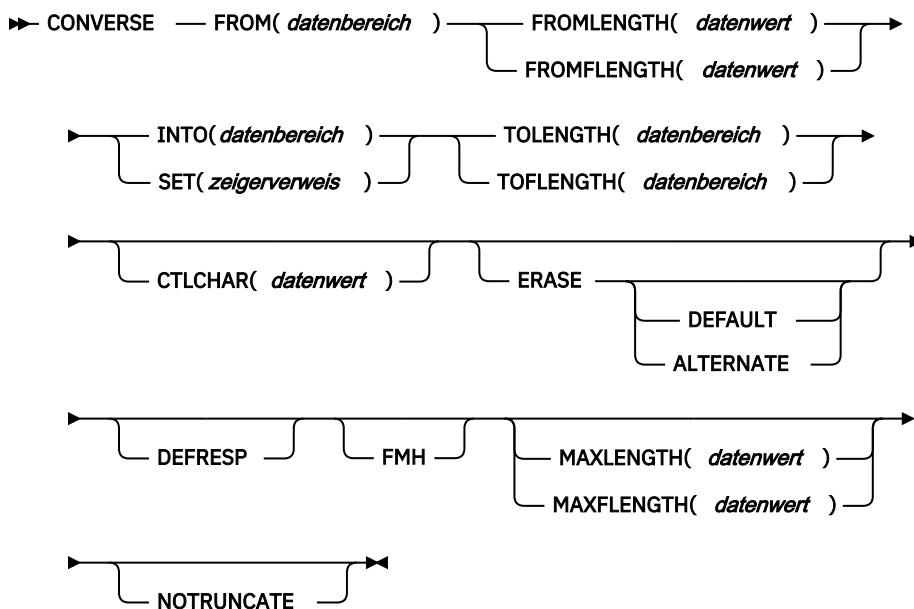
Beschreibung

Der Befehl CONVERSE kommuniziert auf einer logischen Einheit 3650 mit Interpreter.

CONVERSE (3650-3270)

Kommuniziert auf einer logischen Einheit 3650 mit Hostdialog (3270).

CONVERSE (3650-3270)



Bedingungen: LENGERR, TERMERR

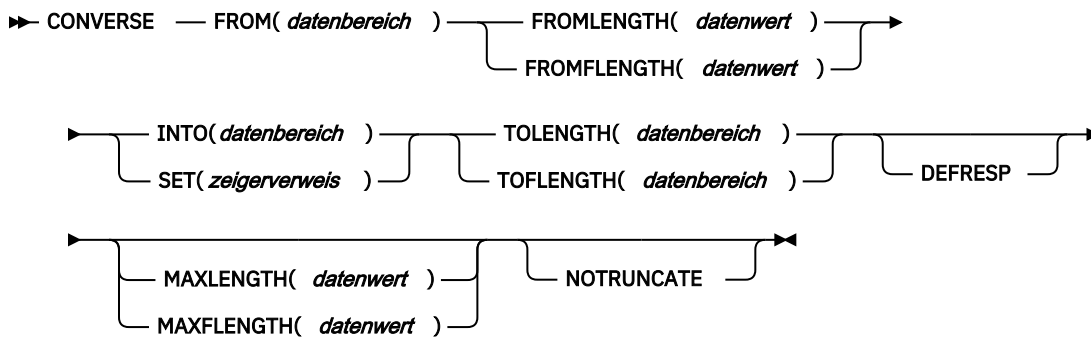
Beschreibung

Der Befehl CONVERSE kommuniziert auf einer logischen Einheit 3650.

CONVERSE (3650-3653)

Kommuniziert auf einer logischen Einheit 3650 mit Hostdialog (3653).

CONVERSE (3650-3653)



Bedingungen: EOC, LENGERR, TERMERR

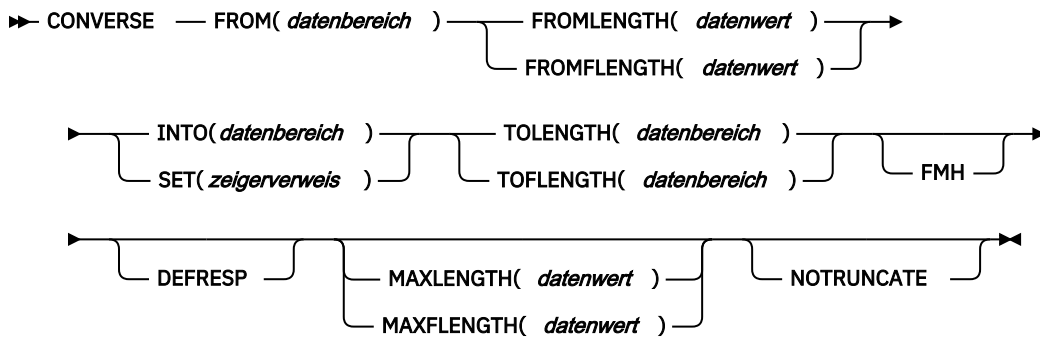
Beschreibung

Der Befehl CONVERSE kommuniziert auf einer logischen Einheit 3650.

CONVERSE (3650-3680)

Kommuniziert auf einer logischen Einheit 3650 mit Hostbefehlsprozessor (3680).

CONVERSE (3650-3680)



Bedingungen: LENGERR, TERMERR

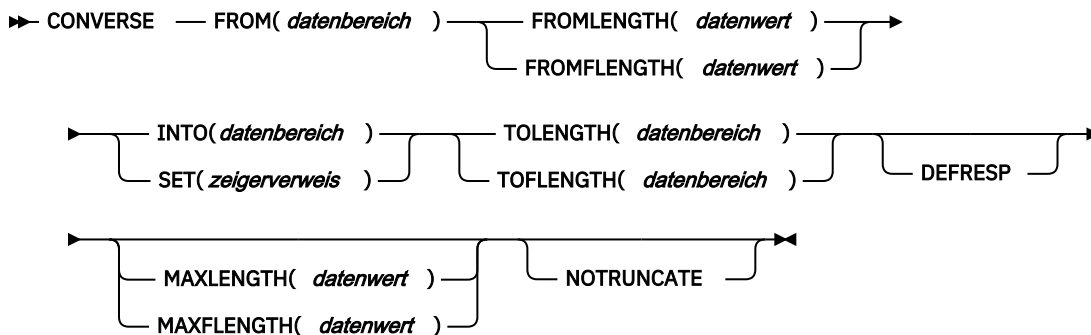
Beschreibung

Der Befehl CONVERSE kommuniziert auf einer logischen Einheit 3650 mit Hostbefehlsprozessor.

CONVERSE (3767)

Kommuniziert auf einer interaktiven logischen Einheit 3767.

CONVERSE (3767)



Bedingungen: EOC, LENGERR, SIGNAL, TERMERR

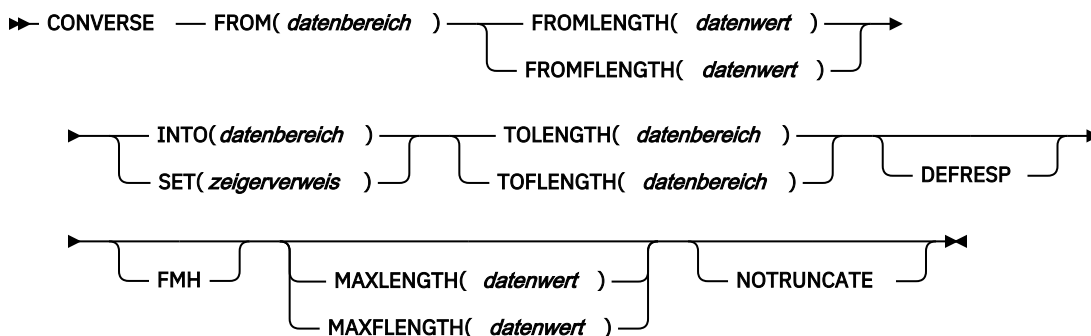
Beschreibung

Der Befehl CONVERSE kommuniziert auf einer interaktiven logischen Einheit 3767. Dieser Befehl gilt außerdem für die interaktive logische Einheit 3770.

CONVERSE (3770)

Kommuniziert auf einer logischen Einheit 3770 mit Stapelverarbeitung.

CONVERSE (3770)



Bedingungen: EOC, EODS, INBFMH, LENGERR, SIGNAL, TERMERR

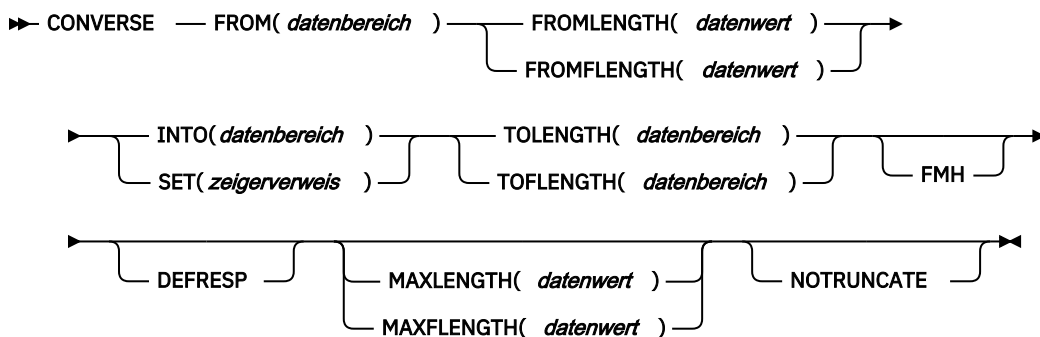
Beschreibung

Der Befehl CONVERSE kommuniziert auf einer logischen Einheit 3770 mit Stapelverarbeitung.

CONVERSE (3790, mit vollem Funktionsumfang oder Anfrage)

Kommuniziert auf einer logischen Einheit 3790 mit vollem Funktionsumfang oder Anfrage.

CONVERSE (3790, mit vollem Funktionsumfang oder Anfrage)



Bedingungen: EOC, EODS, INBFMH, LENGERR, SIGNAL, TERMERR

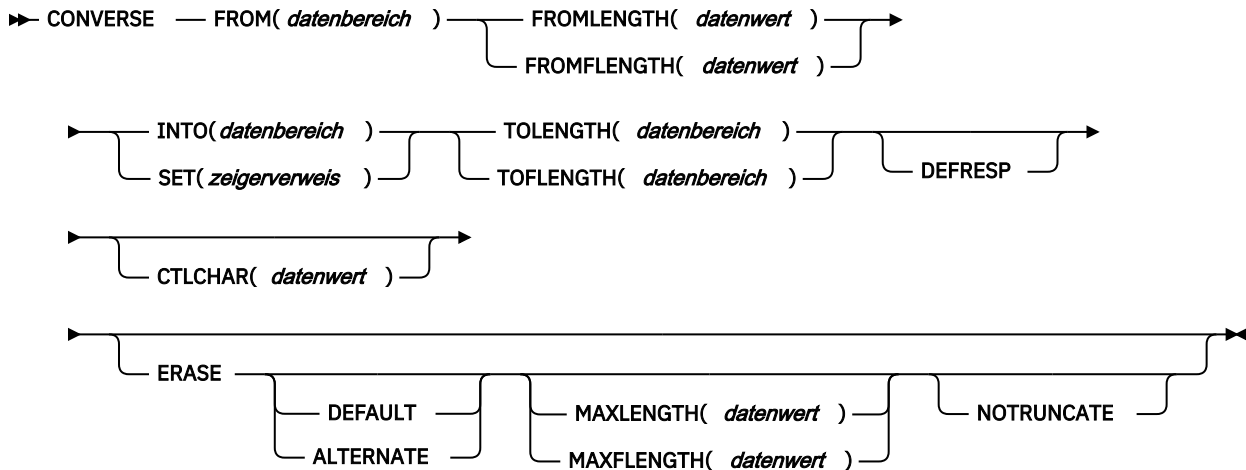
Beschreibung

Der Befehl CONVERSE kommuniziert auf einer logischen Einheit 3790 mit vollem Funktionsumfang oder Anfrage.

CONVERSE (3790- bzw. 3270-Anzeige)

Kommuniziert auf einer logischen Einheit 3790 (3270-Anzeige).

CONVERSE (3790- bzw. 3270-Anzeige)



Bedingungen: LENGERR, TERMERR

Beschreibung

Der Befehl CONVERSE kommuniziert auf einer logische Einheit 3790.

CONVERSE: z/OS Communications Server-Optionen

Allgemeine Optionen, die im Befehl CONVERSE (z/OS Communications Server) verwendet werden.

Optionen

ALTERNATE

Diese Option legt das Terminal auf die Verwendung der alternativen Anzeigegröße fest.

ASIS

Diese Option gibt an, dass die Ausgabe im Transparenzmodus (ohne Erkennung von Steuerzeichen und mit Annahme aller der 256 möglichen Kombinationen von acht Bit als gültige übertragbare Daten).

Anmerkung: Wenn Sie ein Katakana-Terminal verwenden, sehen Sie möglicherweise einige Nachrichten, die gemischte englische Buchstaben und Katakana-Zeichen enthalten. Dies liegt daran, dass Katakana-Terminals keine Ausgaben in gemischter Groß-/Kleinschreibung anzeigen können. Großbuchstaben im Datenstrom werden als englische Großbuchstaben angezeigt, während Kleinbuchstaben als Katakana-Zeichen angezeigt werden. Wenn dies der Fall ist, bitten Sie Ihren Systemprogrammierer, MSGCASE=UPPER in den Systeminitialisierungsparametern anzugeben, sodass Nachrichten nur Großbuchstaben enthalten. Diese Anmerkung gilt für jeden Befehl, der zum Empfang von Katakana-Zeichen verwendet wird, und nicht nur für CONVERSE-Befehle.

ATTACHID(name)

Diese Option gibt an, dass ein Anschlussheader (durch einen Befehl BUILD ATTACH erstellt) den Benutzerdaten, die in der Option FROM angegeben werden, vorausgehen und mit diesen verkettet werden soll. Der Wert "name" (1 - 8 Zeichen) identifiziert den Anschlussheadersteuerblock, der in der lokalen Task verwendet werden soll.

CONVID(name)

Diese Option gibt den Dialog an, auf den sich der Befehl bezieht. Der vier Zeichen lange Name bezeichnet entweder das Token, das von einem zuvor ausgeführten Befehl ALLOCATE in EIBRSRCE im EIB zurückgegeben wurde, oder das Token, das die Hauptsitzung (die durch einen zuvor ausgeführten Befehl ASSIGN zurückgegeben wurde) darstellt.

Aus Gründen der Kompatibilität mit früheren Releases wird SESSION als Synonym für CONVID akzeptiert. Neue Programme verwenden CONVID.

Wenn die Option nicht angegeben wird, wird standardmäßig die Hauptfunktion für die Task verwendet.

CTLCHAR(datenwert)

Diese Option gibt ein 1 Byte großes Schreibbefehlssteuerzeichen (WCC - Write Control Character) an, das den Befehl CONVERSE steuert. Ein COBOL-Benutzer muss einen Datenbereich angeben, der dieses Zeichen enthält.

Wenn die Option nicht angegeben wird, werden alle Tags für modifizierte Daten (MDTs) auf null zurückgesetzt und die Tastatur wird wiederhergestellt.

DEFAULT

Diese Option legt das Terminal auf die Verwendung der Standardanzeigegröße fest.

DEFRESP

Diese Option gibt an, dass eine unbedingte Antwort erforderlich ist, wenn die Ausgabeoperation abgeschlossen ist.

ERASE

Diese Option gibt an, dass der Anzeigedruckerpuffer oder die Partition gelöscht werden soll und der Cursor an die Position in der linken oberen Ecke der Anzeige zurückgestellt werden soll. (Diese Option gilt nur für Einheiten 3270 oder 8775 sowie für das Tastatur- und Anzeigegerät 3604).

Die erste Ausgabeoperation in einer Transaktion oder in einer Reihe von Pseudodialogtransaktionen muss immer ERASE angeben. Für Transaktionen, die an 3270-Anzeigen oder Drucker angeschlossen werden, stellt diese Option, sofern sie nicht explizit durch die Option DEFAULT oder ALTERNATE überschrieben wird, sicher, dass die korrekte Anzeigegröße ausgewählt wird, wie sie für die Transaktion in der Option SCRNSIZE der RDO-Profildefinition definiert ist.

FMH

Diese Option gibt an, dass ein Funktionsverwaltungsheader (FMH - Function Management Header) in die zu schreibenden Daten eingeschlossen wird. Wenn auch die Option ATTACHID angegeben wird, wird das verkettete FMH-Flag im FMH für den Anschluss gesetzt.

Die Verwendung von FMH ist optional und wird nicht von allen Terminaltypen unterstützt. Wenn FMH nicht angegeben wird, führt CICS keine Aktion aus. Ausgenommen hiervon sind 3600/4700-Terminals, für die ein FMH obligatorisch ist. Wenn in diesem Fall FMH nicht angegeben wird, gibt CICS diese Option an und platziert sie in den ersten drei Byte der Nachricht, die Sie zu diesem Zweck reservieren müssen.

FROM(datenbereich)

Diese Option gibt die Daten an, die an das Terminal oder die logische Einheit geschrieben oder an die Partnertransaktion gesendet werden sollen. Diese Option kann in relevanten Fällen weggelassen werden, wenn ATTACHID angegeben wird.

FROMLENGTH(datenwert)

Diese Option ist eine Vollwort-Alternative zu FROMLENGTH.

FROMLENGTH(datenwert)

Diese Option gibt die Länge der Daten als Halbwort-Binärwert an. Eine Beschreibung einer sicheren Obergrenze finden Sie unter [„LENGTH-Optionen in CICS-Befehlen“](#) auf Seite 10.

INTO(datenbereich)

Die Option gibt das empfangende Feld für Daten an, die aus dem Terminal oder der logischen Einheit gelesen werden, oder sie gibt den Zieldatenbereich der Anwendung an, in den Daten aus dem Anwendungsprogramm empfangen werden sollen, das mit dem anderen Ende des aktuellen Dialogs verbunden ist.

LDC(name)

Diese Option gibt die zwei Zeichen lange mnemonische Zeichenfolge an, mit deren Hilfe der numerische Wert des entsprechenden logischen Einheitencodes (LDC) bestimmt wird. Die mnemonische Zeichenfolge bezeichnet einen LDC-Eintrag, der durch ein Makro DFHTCT mit TYPE=LDC definiert wird.

MAXFLENGTH(datenwert)

Diese Option ist eine Vollwort-Alternative zu MAXLENGTH.

MAXLENGTH(datenwert)

Diese Option gibt die maximale Datenmenge (als Halbwort-Binärwert) an, die von CICS als Antwort auf einen Befehl CONVERSE (Standard) wiederhergestellt werden soll. Wenn INTO angegeben wird, überschreibt MAXLENGTH die Verwendung von TOLENGTH als Eingabe für CICS. Wenn SET angegeben wird, bietet MAXLENGTH dem Programm die Möglichkeit, die Datenmenge zu begrenzen, die es auf einmal empfängt.

Falls der angegebene Wert kleiner als Null ist, wird Null angenommen.

Wenn die Länge der Daten den angegebenen Wert überschreitet und die Option NOTRUNCATE nicht angegeben wird, werden die Daten auf die angegebene Länge abgeschnitten und die Bedingung LENGERR tritt auf. Der Datenbereich, der in der Option TOLENGTH angegeben wird, wird auf die ursprüngliche Länge der Daten gesetzt.

Falls die Länge der Daten den angegebenen Wert überschreitet und die Option NOTRUNCATE angegeben wird, behält CICS die übrigen Daten bei und verwendet sie zur Erfüllung nachfolgender Befehle RECEIVE. Der Datenbereich, der in der Option TOLENGTH angegeben wird, wird auf die Länge der zurückgegebenen Daten gesetzt.

Wenn für MAXLENGTH kein Argument codiert wird, nimmt CICS standardmäßig TOLENGTH an.

NOTRUNCATE

Diese Option gibt an, dass die übrigen Daten, wenn die verfügbaren Daten die angeforderte Länge überschreiten, nicht verworfen, sondern beibehalten werden sollen, sodass sie durch nachfolgende Befehle RECEIVE abgerufen werden können.

SESSION(name)

Diese Option gibt die symbolische ID (1 - 4 Zeichen) für einen Terminaleintrag in der Terminalsteuer-tabelle (TCTTE) für die Sitzung an. Diese Option gibt die zu verwendende alternative Funktion an. Wenn weder diese Option noch die Option CONVID angegeben werden, wird die Hauptfunktion für die Task verwendet.

SET(zeigerverweis)

Diese Option gibt den Zeigerverweis an, der auf die Adresse der Daten gesetzt werden soll, die aus dem Terminal gelesen werden. Der Zeigerverweis bleibt bis zum nächsten Befehl CONVERSE (Standard) oder bis zum Ende der Task gültig, sofern er nicht durch andere Befehle oder Anweisungen geändert wird.

Wenn dem Anwendungsprogramm DATALOCATION(ANY) zugeordnet ist, kann die Adresse der Daten ober- oder unterhalb der 16-MB-Grenze liegen.

Falls dem Anwendungsprogramm DATALOCATION(BELOW) zugeordnet ist und sich die Daten oberhalb der 16-MB-Grenze befinden, werden die Daten an eine Adresse unterhalb der 16-MB-Grenze kopiert und die Adresse dieser Kopie wird zurückgegeben.

Wenn für die aktive Task TASKDATAKEY(USER) angegeben wird und der Speicherschutz aktiv ist, werden die Daten in einem Benutzerschlüssel zurückgegeben. Wenn für die aktive Task TASKDATAKEY(CICS) angegeben wird und der Speicherschutz aktiv ist, befinden sich die zurückgegebenen Daten in einem CICS-Schlüssel.

STATE(cvda)

Diese Option ruft den Status des aktuellen Dialogs ab. Die folgenden CVDA-Werte können von CICS zurückgegeben werden:

- ALLOCATED
- CONFFREE
- CONFRECEIVE
- CONFSEND
- FREE
- PENDFREE

- PENDRECEIVE
- RECEIVE
- ROLLBACK
- SEND
- SYNCFREE
- SYNCRECEIVE
- SYNCSEND

STRFIELD

Diese Option gibt an, dass der Datenbereich der Option FROM strukturierte Felder enthält. Wenn diese Option angegeben wird, muss der Inhalt aller strukturierten Felder vom Anwendungsprogramm verarbeitet werden. Der Befehl CONVERSE muss verwendet werden, wenn der angegebene Datenbereich ein strukturiertes Feld einer Lesepartition enthält. (Strukturierte Felder werden im Handbuch [IBM 3270 Data Stream Programmers Reference](#) beschrieben).

Die Optionen CTLCHAR und ERASE schließen die Option STRFIELD aus. Werden sie zusammen mit STRFIELD verwendet, wird eine Fehlnachricht generiert.

TOFLENGTH(datenbereich)

Diese Option ist eine Vollwort-Alternative zu TOLENGTH.

TOLENGTH(datenbereich)

Diese Option gibt die Länge der zu empfangenden Daten als Halbwort-Binärwert an. Wenn Sie die Option INTO angeben, jedoch die Option MAXLENGTH weglassen, gibt "datenbereich" die maximale Länge an, die von dem Programm akzeptiert wird. Wenn der Wert kleiner null ist, wird null angenommen.

Wenn die Länge der Daten den angegebenen Wert überschreitet, jedoch NOTRUNCATE nicht angegeben wurde, werden die Daten auf diesen Wert abgeschnitten und die Bedingung LENGERR tritt auf. Wenn die Daten empfangen wurden, wird der Datenbereich auf die Länge der Daten gesetzt.

Eine Beschreibung einer sicheren Obergrenze finden Sie unter „[LENGTH-Optionen in CICS-Befehlen](#)“ auf Seite 10.

Bedingungen

Einige der folgenden Bedingungen können in Kombination mit anderen auftreten. CICS prüft diese Bedingungen in der folgenden Reihenfolge:

1. EODS
2. INBFMH
3. EOC

Falls mehrere Bedingungen auftreten, wird nur die erste Bedingung an das Anwendungsprogramm übergeben. Allerdings wird im Feld EIBRCODE auf alle aufgetretenen Bedingungen hingewiesen.

62 CBIDERR

Diese Bedingung tritt auf, wenn der angeforderte Anschlussheadersteuerblock, der in der Option AT-TACHID angegeben wurde, nicht gefunden werden kann.

Standardaktion: Die Task wird abnormal beendet.

06 EOC

Diese Bedingung tritt auf, wenn eine Anforderung/Antwort-Einheit (RU) empfangen wird, in der der Anzeiger für Kettenende (EOC - End-of-Chain) gesetzt ist. Das Feld EIBEOC enthält diesen Anzeiger ebenfalls.

Standardaktion: Die Bedingung wird ignoriert.

05 EODS

Diese Bedingung tritt auf, wenn ein Anzeiger für Dateiende (EODS - End-of-Data-set) empfangen wird.

Standardaktion: Die Task wird abnormal beendet.

57 IGREQCD

Diese Bedingung tritt auf, wenn ein Versuch unternommen wird, einen Befehl CONVERSE auszuführen, nachdem ein Datenflusssteuerbefehl SIGNAL mit einem Richtungswechselanforderungscode (RCD - Request Change Direction) von einer logischen Einheit mit LU-Typ 4 (LUTYPE4) empfangen wurde.

Standardaktion: Die Task wird abnormal beendet.

07 INBFMH

Diese Bedingung tritt auf, wenn eine Anforderung/Antwort-Einheit (RU) einen Funktionsverwaltungsheader (FMH) enthält. Das Feld EIBFMH enthält diesen Anzeiger und muss vorzugsweise anstelle des Felds INBFMH verwendet werden. Der Befehl IGNORE CONDITION kann verwendet werden, um die Bedingung zu ignorieren.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

RESP2-Werte:

200

Eine Distributed Program Link-Serveranwendung hat die Funktionsverlagerungssitzung (ihre Hauptfunktion) angegeben.

Diese Bedingung tritt auch in einer der folgenden Situationen auf (ohne RESP2-Wert):

- Der Befehl wird in einem Dialog verwendet, der zurzeit von CPI Communications verwendet wird oder bei dem es sich um einen APPC-Basisdialog handelt. Im letzteren Fall setzt die Anwendung einen Befehl GDS SEND INVITE, gefolgt von einem Befehl GDS RECEIVE, ab.

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

Diese Bedingung tritt in einer der folgenden Situationen auf:

- Daten, die empfangen wurden, wurden von CICS verworfen, weil ihre Länge die maximale Länge überschreitet, die vom Programm akzeptiert wird (siehe Optionen TOLENGTH und MAXLENGTH), und die Option NOTRUNCATE nicht angegeben wurde.
- Ein Wert außerhalb des gültigen Bereichs wurde in einer der folgenden Optionen angegeben: FROMLENGTH, FROMFLENGTH, MAXLENGTH, MAXFLENGTH, TOLENGTH oder TOFLENGTH.

Standardaktion: Die Task wird abnormal beendet.

61 NOTALLOC

Diese Bedingung tritt auf, wenn die im Befehl angegebene Funktion (Facility) nicht der Anwendung gehört oder sich nicht auf einen Dialog bezieht, der der Anwendung gehört.

Standardaktion: Die Task wird abnormal beendet.

24 SIGNAL

Diese Bedingung tritt auf, wenn ein eingehender Datenflusssteuerbefehl SIGNAL von einer logischen Einheit oder von einer Sitzung oder von der Partnertransaktion empfangen wird. Das Feld EIBSIG wird immer gesetzt, wenn ein eingehendes Signal empfangen wird.

Standardaktion: Die Bedingung wird ignoriert.

81 TERMERR

Diese Bedingung tritt für einen terminal- oder sitzungsbezogenen Fehler auf. Jede Aktion außer einem Befehl FREE für diesen Dialog verursacht einen Abbruch mit dem Code ATCV.

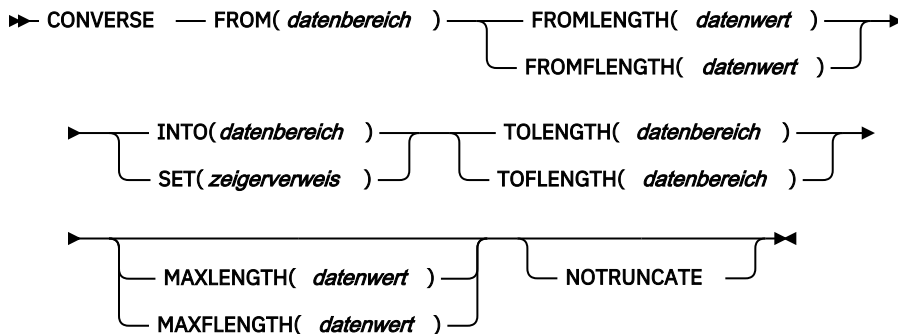
Eine Anforderung CANCEL TASK durch ein Benutzerknotenfehlerprogramm (NEP) kann diese Bedingung auslösen, wenn die Task eine ausstehende aktive Terminalsteueranforderung hat, wenn das Programm für abnormale Knotenbedingungen den Sitzungsfehler verarbeitet.

Standardaktion: Die Task wird abnormal mit dem Abbruchcode ATNI beendet.

CONVERSE (Nicht-z/OS Communications Server-Standard)

Kommuniziert in der CICS-Standardterminalunterstützung.

CONVERSE (Standard)



Bedingungen: LENGERR

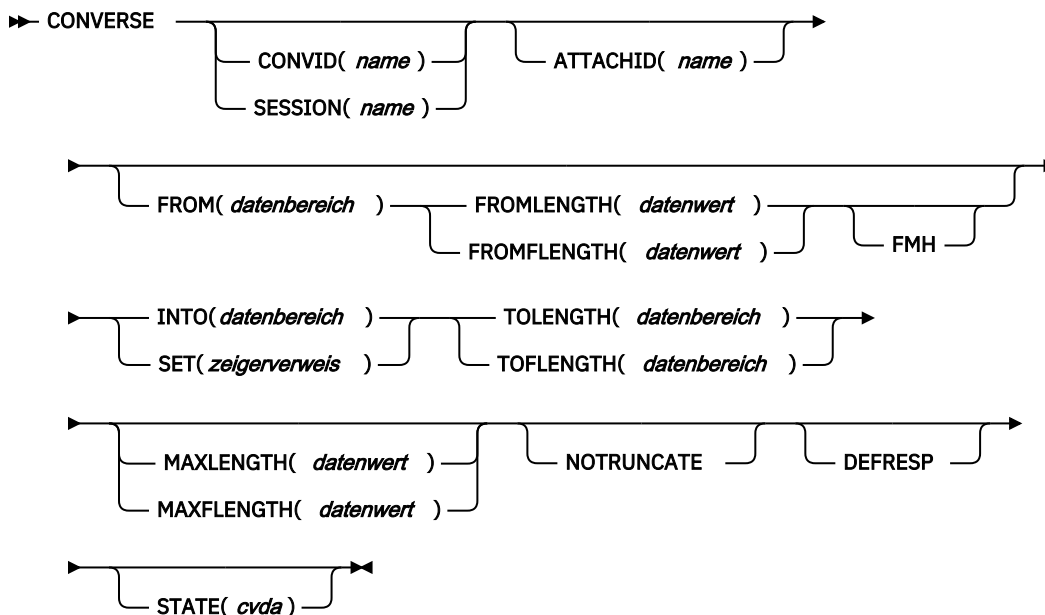
Beschreibung

Diese Form des Befehls CONVERSE wird von allen von CICS unterstützten Terminals verwendet, für die die anderen CONVERSE-Beschreibungen nicht geeignet sind.

CONVERSE (MRO)

Kommuniziert in einer MRO-Sitzung.

CONVERSE (MRO)



Bedingungen: CBIDERR, EOC, INBFMH, LENGERR, NOTALLOC, TERMERR

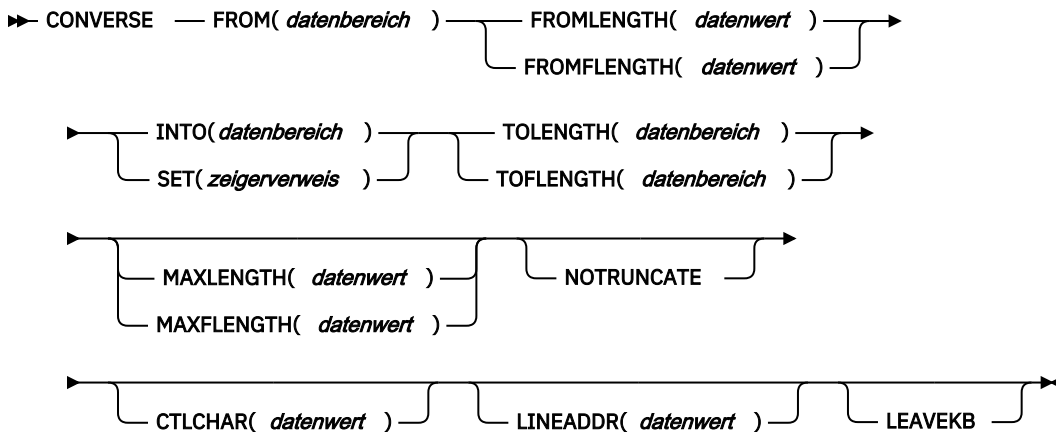
Beschreibung

Der Befehl CONVERSE kommuniziert in einer MRO-Sitzung. Weitere Informationen zu MRO und IRC finden Sie unter [Einführung in die übergreifende Kommunikation in CICS](#).

CONVERSE (2260)

Kommuniziert auf einer Anzeigestation (Datensichtgerät) 2260 oder 2265.

CONVERSE (2260)



Bedingung: LENGERR

Beschreibung

Der Befehl CONVERSE kommuniziert auf einer Anzeigestation (Datensichtgerät) 2260 oder 2265.

CONVERSE: Nicht-z/OS Communications Server-Optionen

Allgemeine Optionen, die im Befehl CONVERSE (Nicht-z/OS Communications Server) verwendet werden.

Optionen

ALTERNATE

Diese Option legt das Terminal auf die Verwendung der alternativen Anzeigegröße fest.

ASIS

Diese Option gibt an, dass die Ausgabe im Transparenzmodus (ohne Erkennung von Steuerzeichen und mit Annahme aller der 256 möglichen Kombinationen von 8 Bit als gültige übertragbare Daten).

Anmerkung: Wenn Sie ein Katakana-Terminal verwenden, sehen Sie möglicherweise einige Nachrichten, die gemischte englische Buchstaben und Katakana-Zeichen enthalten. Dies liegt daran, dass Katakana-Terminals keine Ausgaben in gemischter Groß-/Kleinschreibung anzeigen können. Großbuchstaben im Datenstrom werden als englische Großbuchstaben angezeigt, während Kleinbuchstaben als Katakana-Zeichen angezeigt werden. Bitten Sie in einem solchen Fall Ihren Systemprogrammierer, in den Systeminitialisierungsparametern MSGCASE=UPPER anzugeben, damit diese Nachrichten ausschließlich Großbuchstaben enthalten.

ATTACHID(name)

Diese Option gibt an, dass ein Anschlussheader (durch einen Befehl BUILD ATTACH erstellt) den Benutzerdaten, die in der Option FROM angegeben werden, vorausgehen und mit diesen verkettet werden soll. Der Wert "name" (1-8 Zeichen) identifiziert den Anschlussheadersteuerblock, der in der lokalen Task verwendet werden soll.

CONVID(name)

Diese Option gibt den Dialog an, auf den sich der Befehl bezieht. Der vier Zeichen lange Name bezeichnet entweder das Token, das von einem zuvor ausgeführten Befehl ALLOCATE in EIBRSRCE im EIB zurückgegeben wurde, oder das Token, das die Hauptsitzung (die durch einen zuvor ausgeführten Befehl ASSIGN zurückgegeben wurde) darstellt.

CTLCHAR(datenwert)

Diese Option gibt ein 1 Byte großes Schreibbefehlssteuerzeichen (WCC - Write Control Character) an, das den Befehl CONVERSE steuert. (Das Schreibbefehlssteuerzeichen (WCC) ist in [IBM 3270 Data](#)

Stream Programmers Reference dokumentiert. Ein COBOL-Benutzer muss einen Datenbereich angeben, der dieses Zeichen enthält. Wenn die Option nicht angegeben wird, werden alle Tags für modifizierte Daten (MDTs) auf null zurückgesetzt und die Tastatur wird wiederhergestellt.

DEFAULT

Diese Option legt das Terminal auf die Verwendung der Standardanzeigegröße fest.

DEFRESP

Diese Option gibt an, dass eine unbedingte Antwort erforderlich ist, wenn die Ausgabeoperation abgeschlossen ist.

ERASE

Diese Option gibt an, dass der Anzeigedruckerpuffer oder die Partition gelöscht werden soll und der Cursor an die Position in der linken oberen Ecke der Anzeige zurückgestellt werden soll. (Diese Option gilt nur für Einheiten 3270 oder 8775 sowie für das Tastatur- und Anzeigegerät 3604).

Die erste Ausgabeoperation in einer Transaktion oder in einer Reihe von Pseudodialogtransaktionen muss immer ERASE angeben. Für Transaktionen, die an 3270-Anzeigen oder Drucker angeschlossen werden, stellt diese Option, sofern sie nicht explizit durch die Option DEFAULT oder ALTERNATE überschrieben wird, sicher, dass die korrekte Anzeigegröße ausgewählt wird, wie sie für die Transaktion in der Option SCRNSIZE der RDO-Profildefinition definiert ist.

FMH

Diese Option gibt an, dass ein Funktionsverwaltungsheader (FMH - Function Management Header) in die zu schreibenden Daten eingeschlossen wird. Wenn auch die Option ATTACHID angegeben wird, wird das verkettete FMH-Flag im FMH für den Anschluss gesetzt.

FROM(datenbereich)

Diese Option gibt die Daten an, die an das Terminal oder die logische Einheit geschrieben oder an die Partnertransaktion gesendet werden sollen. Diese Option kann in relevanten Fällen weggelassen werden, wenn ATTACHID angegeben wird.

FROMLENGTH(datenwert)

Diese Option ist eine Vollwort-Alternative zu FROMLENGTH.

FROMLENGTH(datenwert)

Diese Option gibt die Länge der zu schreibenden Daten als Halbwort-Binärwert an. Wenn Sie diese Option verwenden, müssen Sie auch FROM angeben. Eine Beschreibung einer sicheren Obergrenze finden Sie unter „LENGTH-Optionen in CICS-Befehlen“ auf Seite 10.

INTO(datenbereich)

Die Option gibt das empfangende Feld für die Daten an, die aus der logischen Einheit oder aus dem Terminal gelesen werden.

LEAVEKB

Diese Option gibt an, dass die Tastatur nach Abschluss der Datenübertragung gesperrt bleiben soll.

LINEADDR(datenwert)

Diese Option gibt an, dass das Schreiben in einer bestimmten Zeile einer 2260/2265-Anzeige beginnen soll. Der Datenwert ist ein Halbwort-Binärwert aus dem Bereich von 1 - 12 für eine 2260-Anzeige oder aus dem Bereich von 1 - 15 für eine 2265-Anzeige.

MAXLENGTH(datenwert)

Diese Option ist eine Vollwort-Alternative zu MAXLENGTH.

MAXLENGTH(datenwert)

Diese Option gibt die maximale Datenmenge (als Halbwort-Binärwert) an, die von CICS als Antwort auf einen Befehl CONVERSE wiederhergestellt werden soll. Wenn INTO angegeben wird, überschreibt MAXLENGTH die Verwendung von TOLENGTH als Eingabe für CICS. Wenn SET angegeben wird, bietet MAXLENGTH dem Programm die Möglichkeit, die Datenmenge zu begrenzen, die es auf einmal empfängt.

Falls der angegebene Wert kleiner als Null ist, wird Null angenommen.

Wenn die Länge der Daten den angegebenen Wert überschreitet und die Option NOTTRUNCATE nicht angegeben wird, werden die Daten auf die angegebene Länge abgeschnitten und die Bedingung LEN-

GERR tritt auf. Der Datenbereich, der in der Option TOLength angegeben wird, wird auf die ursprüngliche Länge der Daten gesetzt.

Falls die Länge der Daten den angegebenen Wert überschreitet und die Option NOTRUNCATE angegeben wird, behält CICS die übrigen Daten bei und verwendet sie zur Erfüllung nachfolgender Befehle RECEIVE. Der Datenbereich, der in der Option TOLength angegeben wird, wird auf die Länge der zurückgegebenen Daten gesetzt.

Wenn für MAXLENGTH kein Argument codiert wird, nimmt CICS standardmäßig TOLength an.

NOTRUNCATE

Diese Option gibt an, dass die übrigen Daten, wenn die verfügbaren Daten die angeforderte Länge überschreiten, nicht verworfen, sondern beibehalten werden sollen, sodass sie durch nachfolgende Befehle RECEIVE abgerufen werden können.

PSEUDOBIN

Diese Option gibt an, dass die Daten, die gelesen und geschrieben werden, aus der pseudobinären System/7-Darstellung in die hexadezimale Darstellung konvertiert werden sollen.

SESSION(name)

Diese Option gibt die symbolische ID (1 - 4 Zeichen) für einen Terminaleintrag in der Terminalsteuer-tabelle (TCTTE) für die Sitzung an. Diese Option gibt die alternative Funktion an, die verwendet werden soll. Wenn weder diese Option noch die Option CONVID angegeben werden, wird die Hauptfunktion für die Task verwendet.

SET(zeigerverweis)

Diese Option gibt einen Zeigerverweis an, der auf die Adresse von Daten gesetzt werden soll, die vom Dialogpartner in einem MRO-Dialog empfangen werden. Der Zeigerverweis bleibt bis zum nächsten Befehl CONVERSE (MRO) oder bis zum Ende der Task gültig, sofern er nicht durch andere Befehle oder Anweisungen geändert wird.

Wenn dem Anwendungsprogramm DATALOCATION(ANY) zugeordnet ist, kann die Adresse der Daten ober- oder unterhalb der 16-MB-Grenze liegen.

Wenn dem Anwendungsprogramm DATALOCATION(BELOW) zugeordnet ist und sich die Daten oberhalb der 16-MB-Grenze befinden, werden die Daten an eine Adresse unterhalb der 16-MB-Grenze kopiert und die Adresse dieser Kopie wird zurückgegeben.

Wenn für die aktive Task TASKDATAKEY(USER) angegeben wird und der Speicherschutz aktiv ist, befinden sich die zurückgegebenen Daten in einem Benutzerschlüssel. Wenn für die aktive Task TASKDATAKEY(CICS) angegeben wird und der Speicherschutz aktiv ist, befinden sich die zurückgegebenen Daten in einem CICS-Schlüssel.

STATE(cvda)

Diese Option ruft den Status des Transaktionsprogramms ab. Die folgenden CVDA-Werte können von CICS zurückgegeben werden:

- ALLOCATED
- FREE
- PENDFREE
- RECEIVE
- ROLLBACK
- SEND
- SYNCFREE
- SYNCRECEIVE
- SYNCSEND

STRFIELD

Diese Option gibt an, dass der Datenbereich, der in der Option FROM angegeben wird, strukturierte Felder enthält. Wenn diese Option angegeben wird, muss der Inhalt aller strukturierten Felder vom Anwendungsprogramm verarbeitet werden. Anstelle des Befehls SEND muss der Befehl CONVERSE

verwendet werden, falls der Datenbereich ein strukturiertes Feld für eine Lesepartition enthält. (Strukturierte Felder werden im Handbuch [IBM 3270 Data Stream Programmers Reference](#) beschrieben.)

Die Optionen CTLCHAR und ERASE schließen die Option STRFIELD aus. Werden sie zusammen mit STRFIELD verwendet, wird eine Fehlermeldung generiert.

TOFLENGTH(datenbereich)

Diese Option ist eine Vollwort-Alternative zu TOLENGTH.

TOLENGTH(datenbereich)

Diese Option gibt die Länge der zu empfangenden Daten als Halbwort-Binärwert an. Wenn Sie die Option INTO angeben, jedoch die Option MAXLENGTH weglassen, gibt "datenbereich" die maximale Länge an, die von dem Programm akzeptiert wird. Wenn der Wert kleiner null ist, wird null angenommen.

Wenn die Länge der Daten den angegebenen Wert überschreitet, jedoch NOTRUNCATE nicht angegeben wurde, werden die Daten auf diesen Wert abgeschnitten und die Bedingung LENGERR tritt auf. Wenn die Daten empfangen wurden, wird der Datenbereich auf die Länge der Daten gesetzt.

Eine Beschreibung einer sicheren Obergrenze finden Sie unter [„LENGTH-Optionen in CICS-Befehlen“](#) auf Seite 10.

Bedingungen

Einige der folgenden Bedingungen können in Kombination mit anderen auftreten. CICS prüft diese Bedingungen in der folgenden Reihenfolge:

1. INBFMH
2. EOC

Falls mehrere Bedingungen auftreten, wird nur die erste Bedingung an das Anwendungsprogramm übergeben. Allerdings wird im Feld EIBRCODE auf alle aufgetretenen Bedingungen hingewiesen.

62 CBIDERR

Diese Bedingung tritt auf, wenn der angeforderte Anschlussheadersteuerblock, der in der Option ATTACHID angegeben wurde, nicht gefunden werden kann.

Standardaktion: Die Task wird abnormal beendet.

06 EOC

Diese Bedingung tritt auf, wenn eine Anforderung/Antwort-Einheit (RU) empfangen wird, in der der Anzeiger für Kettenende (EOC - End-of-Chain) gesetzt ist. Das Feld EIBEOC enthält diesen Anzeiger ebenfalls.

Standardaktion: Die Bedingung wird ignoriert.

04 EOF

Diese Bedingung tritt auf, wenn ein Anzeiger für Dateiende (EOF - End-of-File) empfangen wird.

Standardaktion: Die Task wird abnormal beendet.

07 INBFMH

Diese Bedingung tritt auf, wenn eine Anforderung/Antwort-Einheit (RU) einen Funktionsverwaltungsholder (FMH) enthält. Das Feld EIBFMH enthält diesen Anzeiger und sollte vorzugsweise anstelle des Felds INBFMH verwendet werden. Der Befehl IGNORE CONDITION kann verwendet werden, um die Bedingung zu ignorieren.

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

Diese Bedingung tritt in einer der folgenden Situationen auf:

- Daten werden von CICS verworfen, weil ihre Länge die maximale Länge überschreitet, die vom Programm akzeptiert wird und die Option NOTRUNCATE nicht angegeben wurde.
- Ein Wert außerhalb des gültigen Bereichs wurde in der Option FROMLENGTH angegeben.

Standardaktion: Die Task wird abnormal beendet.

61 NOTALLOC

Diese Bedingung tritt auf, wenn die im Befehl angegebene Funktion (Facility) nicht der Anwendung gehört.

Standardaktion: Die Task wird abnormal beendet.

02 RDATT

Diese Bedingung tritt auf, wenn der Empfangsteil ("receive") des Dialogs durch die Abruftaste (ATTN) und nicht durch die Rückföhrtaste (Return) beendet wird.

Standardaktion: Die Bedingung wird ignoriert.

81 TERMERR

Diese Bedingung tritt für einen sitzungsbezogenen Fehler auf.

Eine Anforderung CANCEL TASK durch ein Benutzerknotenfehlerprogramm (NEP) kann diese Bedingung auslösen, wenn die Task eine ausstehende aktive Terminalsteueranforderung hat, wenn das Programm für abnormale Knotenbedingungen den Sitzungsfehler verarbeitet.

Standardaktion: Die Task wird abnormal mit dem Abbruchcode ATNI beendet.

03 WRBRK

Diese Bedingung tritt auf, wenn der Sendeteil ("send") des Dialogs durch die Abruftaste (ATTN) und nicht durch die Rückföhrtaste (Return) beendet wird.

Standardaktion: Die Bedingung wird ignoriert.

CONVERTTIME

Konvertiert eine konzipierte Datums- und Zeitmarkenzeichenfolge in das Format ABSTIME.

CONVERTTIME

➡ CONVERTTIME — DATESTRING(*datenbereich*) — ABSTIME(*datenbereich*) ➡

Bedingung: INVREQ

Dieser Befehl ist threadsicher.

Beschreibung

CONVERTTIME analysiert vier verschiedene Datums- und Zeitmarkenformate, die häufig im Internet verwendet werden, und konvertiert sie in das ABSTIME-Format (absolutes Datums- und Zeitformat) in Ortszeit.

Das ABSTIME-Format gibt die Zeit im gepackten Dezimalformat seit dem 01.01.1900 00:00 Uhr an. Die Zeit wird in Millisekunden angegeben und wird immer abgeschnitten, nie gerundet. Der Befehl FORMATTIME kann dazu verwendet werden, diese Daten in andere Formate zu ändern.

Die folgenden konzipierten Datums- und Zeitmarkenzeichenfolgeformate werden vom Befehl CONVERTTIME erkannt:

RFC 1123-Format

Das bevorzugte Standardformat für Datums- und Zeitmarken für das HTTP-Protokoll, wie in [RFC 1123](#) spezifiziert. Ein Beispiel für eine Datums- und Zeitmarke in diesem Format ist "Tue, 01 Apr 2003 10:01:02 +0000".

RFC 3339-Format

Der XML-Datentyp 'dateTime', der in [RFC 3339](#) spezifiziert ist und dem Standard ISO 8601 entnommen wurde. Ein Beispiel für eine Datums- und Zeitmarke in diesem Format ist "2003-04-01T10:01:02.498Z". Datums- und Zeitmarken in diesem Format sind in UTC (Coordinated Universal Time, koordinierte Weltzeit). Der Zeitonenunterschied (-12:00 bis +12:00) wird am Ende der Datums- und Zeitmarke angegeben oder es wird der Buchstabe Z für keinen Unterschied (+00:00) angegeben. Der Dezimalbruch einer Sekunde, der in diesem Beispiel gezeigt wird, ist optional.

RFC 850-Format

Ein älteres Datums- und Zeitmarkenformat für das Internet, das in RFC 850 spezifiziert ist. Ein Beispiel für eine Datums- und Zeitmarke in diesem Format ist "Tuesday, 01-Apr-03 10:01:02 GMT".

Wichtig: Da das Jahr in diesem Format nur zwei Stellen hat, geht CICS von der Annahme aus, dass Jahre im Bereich von 1970 bis 2069 liegen. Im obigen Beispiel nimmt CICS an, dass das Datum des Dokuments der 1. April 2003 ist. Für die Datums- und Zeitmarke "Thursday, 13-Feb-98 15:30:00 GMT", nimmt CICS an, dass das Datum des Dokuments der 13. Februar 1998 ist. Berücksichtigen Sie dieses Verhalten bei der Codierung Ihrer Anwendung, wenn Sie davon ausgehen, dass Sie Datums- und Zeitmarken in diesem Format empfangen könnten.

ASCTime-Format

Ein Datums- und Zeitmarkenformat, das von der C-Funktion ASCTime erzeugt wird. Ein Beispiel für eine Datums- und Zeitmarke in diesem Format ist "Tue Apr 1 10:01:02 2003".

Optionen

DATESTRING(*datenbereich*)

Diese Option gibt einen 64 Zeichen langen Datenbereich an, der die konzipierte Datums- und Zeitmarkenzeichenfolge aufnehmen soll. Sie können eine Zeichenfolge in einem der Formate angeben, die von diesem Befehl erkannt werden. Wenn die Zeichenfolge weniger als 64 Zeichen lang ist, füllen Sie sie mit Leerzeichen oder Nullen auf. Sie müssen das Format der Daten in der Option DATESTRING nicht angeben, da CICS die Daten automatisch liest, um zu bestimmen, ob sie ein unterstütztes Format haben. Das Datum und die Zeit werden für den Wert in ABSTIME, der zurückgegeben wird, in Ortszeit umgewandelt.

ABSTIME(*datenbereich*)

Diese Option gibt einen Datenbereich an, der die konvertierte Datums- und Zeitmarke im ABSTIME-Format empfangen soll. Wenn die Datums- und Zeitmarke kein erkanntes Format hat, wird im ABSTIME-Datenbereich null zurückgegeben.

Bedingungen

16 INVREQ

Die RESP2-Werte lauten wie folgt:

1

Das Format der Datums- und Zeitmarkenzeichenfolge wurde nicht als eines der Formate erkannt, die von diesem Befehl unterstützt werden. Dieser Fehler kann durch eine Datums- und Zeitmarkenzeichenfolge verursacht werden, die ein unterstütztes Format hat, jedoch Formatfehler enthält. Beispiel: Ein Jahreswert, der mehr oder weniger als die richtige Anzahl von Stellen für das Format hat, oder ein Element, das numerisch sein müsste, dies jedoch nicht ist.

2

Ungültige Zeit.

3

Ungültiger Monat.

4

Ungültiges Jahr (schließt Jahre vor 1900 ein).

5

Ungültiger Tagesname.

6

Ungültige Tageszahl für den angegebenen Monat und das angegebene Jahr.

7

GMT wurde nicht angegeben (für RFC 850-Formate erforderlich).

8

Ungültiger Bruchteil einer Sekunde.

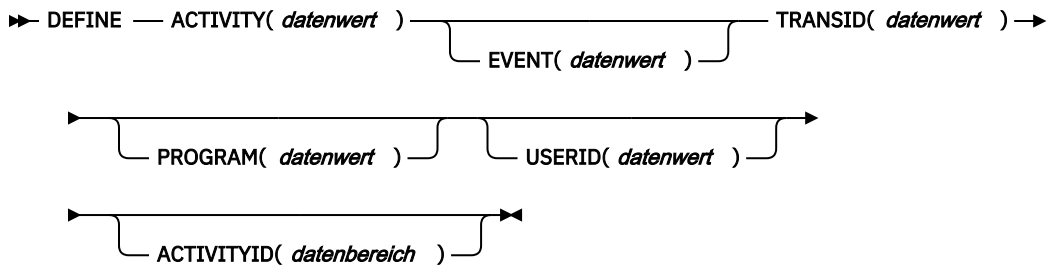
9

Ungültiger Wert für den Zeitzoneunterschied.

DEFINE ACTIVITY

Definiert eine CICS-BTS-Aktivität.

DEFINE ACTIVITY



Bedingungen: ACTIVITYERR, EVENTERR, INVREQ, IOERR, NOTAUTH, TRANSIDERR

Beschreibung

Der Befehl DEFINE ACTIVITY definiert eine Aktivität für CICS Business Transaction Services (BTS). Mit diesem Befehl kann der aktuellen Aktivität eine untergeordnete Aktivität hinzugefügt werden.

Der Name des Programms, das in der Ausführung der neuen Aktivität verwendet wird, wird entweder der Option PROGRAM entnommen oder, falls PROGRAM nicht angegeben wird, der Transaktionsdefinition entnommen, auf die die Option TRANSID verweist.

Die Transaktionsattribute, die in den Optionen TRANSID und USERID angegeben werden, werden wirksam, wenn die Aktivität durch einen Befehl RUN aktiviert wird, jedoch *nicht*, wenn sie durch einen Befehl LINK aktiviert wird (siehe „Kontextwechsel“ auf Seite 477).

BTS schreibt das Hinzufügen der Aktivität erst fest, wenn die anfordernde Transaktion einen erfolgreichen Synchronisationspunkt erfasst hat.

Optionen

ACTIVITY(datenwert)

Gibt den Namen (1-16 Zeichen) der neuen Aktivität an. Der Name darf nicht der Name einer anderen untergeordneten Aktivität der Aktivität sein, die den Befehl DEFINE absetzt.

Zulässige Zeichen: A-Z a-z 0-9 \$ @ # / % & ? ! : | " = ~ , ; < > . - und _. Führende und eingebettete Leerzeichen sind nicht zulässig. Wenn der angegebene Name kürzer als 16 Zeichen ist, wird er mit nachfolgenden Leerzeichen bis auf 16 Zeichen aufgefüllt.

ACTIVITYID(datenbereich)

Gibt die 52 Zeichen lange ID an, die von CICS der neu definierten Aktivität zugeordnet wird. Diese ID ist im gesamten Sysplex eindeutig.

EVENT(datenwert)

Gibt den Namen (1-16 Zeichen) des Beendigungsereignisses für die Aktivität an. Das Beendigungsereignis wird an die übergeordnete Aktivität der Aktivität gesendet, wenn die Aktivität beendet wird.

Wenn EVENT nicht angegeben wird, erhält das Beendigungsereignis den gleichen Namen wie die Aktivität selbst.

Zulässige Zeichen: A-Z a-z 0-9 \$ @ # . - und _. Führende und eingebettete Leerzeichen sind nicht zulässig. Wenn der angegebene Name kürzer als 16 Zeichen ist, wird er mit nachfolgenden Leerzeichen bis auf 16 Zeichen aufgefüllt.

PROGRAM(datenwert)

Gibt den Namen (1-8 Zeichen) des Programms für die Aktivität an, die definiert wird. Wenn kein Programm angegeben wird, wird der Name aus der TRANSID-Definition genommen.

TRANSID(datenwert)

Gibt den Namen (1-4 Zeichen) der Transaktion an, unter der die Aktivität ausgeführt werden soll, wenn sie durch einen Befehl RUN aktiviert wird.

Anmerkung: Wenn die Aktivität durch einen Befehl LINK aktiviert wird, wird sie unter der Transaktions-ID (TRANSID) der Transaktion ausgeführt, die den Befehl LINK absetzt.

Die Transaktion muss in der CICS-Region definiert sein, in der der Prozess ausgeführt wird.

USERID(datenwert)

Gibt die Benutzer-ID (1-8 Zeichen) an, unter deren Berechtigung die Aktivität ausgeführt werden soll, wenn sie durch einen Befehl RUN aktiviert wird.

Anmerkung: Wenn die Aktivität durch einen Befehl LINK aktiviert wird, wird sie unter der Benutzer-ID (USERID) der Transaktion ausgeführt, die den Befehl LINK absetzt.

Der Wert dieses Felds wird als *definierte Benutzer-ID* bezeichnet.

Wenn Sie die Option USERID nicht angeben, wird als definierte Benutzer-ID standardmäßig die Benutzer-ID verwendet, unter der die Transaktion, die den Befehl DEFINE absetzt, ausgeführt wird. Diese Benutzer-ID wird als *Befehlsbenutzer-ID* bezeichnet.

Wird die Option USERID angegeben, führt CICS (zum Zeitpunkt der Definition) eine Ersatzsicherheitsprüfung durch, um sicherzustellen, dass die Befehlsbenutzer-ID berechtigt ist, die definierte Benutzer-ID zu verwenden. Wenn Sie also die Option USERID angeben, müssen Sie die Befehlsbenutzer-ID als Ersatzbenutzer der definierten Benutzer-ID berechnen.

Bedingungen**109 ACTIVITYERR**

RESP2-Werte:

3

Der in der Option ACTIVITY angegebene Name wurde bereits verwendet, um eine andere untergeordnete Aktivität der aktuellen Aktivität zu benennen.

111 EVENTERR

RESP2-Werte:

7

Das Beendigungsereignis, das in der Option EVENT angegeben wurde, wurde im Ereignispool der aktuellen Aktivität bereits definiert.

16 INVREQ

RESP2-Werte:

4

Der Befehl DEFINE ACTIVITY wurde außerhalb des Gültigkeitsbereichs einer momentan aktiven Aktivität abgesetzt.

17

Der in der Option ACTIVITY angegebene Aktivitätsname oder der in der Option EVENT angegebene Ereignisname ist ungültig.

17 IOERR

RESP2-Werte:

29

Die Repository-Datei ist nicht verfügbar.

30

Es ist ein Ein-/Ausgabefehler für die Repository-Datei aufgetreten.

70 NOTAUTH

RESP2-Werte:

101

Der Benutzer, der der absetzenden Task zugeordnet ist, ist nicht berechtigt, auf die Datei zuzugreifen, die der BTS-Repository-Datei zugeordnet ist, in der Details der Aktivität gespeichert werden sollen.

102

Der Benutzer, der der absetzenden Task zugeordnet ist, ist nicht als Ersatz für die definierte Benutzer-ID berechtigt, die in der Option USERID angegeben wurde.

28 TRANSIDERR

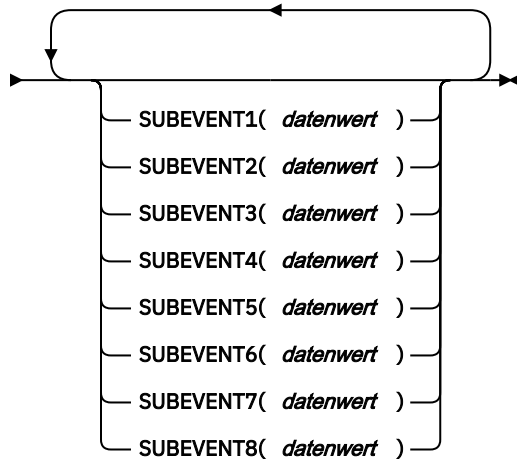
Die in der Option TRANSID angegebene Transaktions-ID ist für CICS nicht definiert.

DEFINE COMPOSITE EVENT

Definiert ein zusammengesetztes Ereignis.

DEFINE COMPOSITE EVENT

➡ DEFINE — COMPOSITE — EVENT(*datenwert*) — AND — OR —



Bedingungen: EVENTERR, INVREQ

Beschreibung

Der Befehl DEFINE COMPOSITE EVENT definiert ein zusammengesetztes Ereignis für Business Transaction Services (BTS). Ein zusammengesetztes Ereignis wird aus null oder mehr atomaren Ereignissen zusammengesetzt, die als Unterereignisse bezeichnet werden.

Der Befehl DEFINE COMPOSITE EVENT definiert ein Vergleichselement (*Prädikat*), bei dem es sich um einen logischen Ausdruck handelt, der Unterereignisse einbezieht. Zu allen Zeitpunkten gibt der Ausführungsstatus (FIRED oder NOTFIRED) des zusammengesetzten Ereignisses den Wert des Prädikats wieder. Wenn das Prädikat wahr wird, wird das zusammengesetzte Ereignis ausgelöst. Wenn es falsch wird, wird der Ausführungsstatus des zusammengesetzten Ereignisses auf NOTFIRED zurückgesetzt.

Der logische Operator, der auf die Unterereignisse im Prädikat des zusammengesetzten Ereignisses angewendet wird, ist einer der booleschen Operatoren AND oder OR. *Es können nicht sowohl AND als auch OR verwendet werden.*

Sie können bis zu acht Unterereignisse angeben, die dem zusammengesetzten Ereignis hinzugefügt werden sollen, wenn das zusammengesetzte Ereignis erstellt wird. Wenn Sie keine Unterereignisse angeben, wird das zusammengesetzte Ereignis als "leer" definiert, das heißt, ohne enthaltene Unterverzeichnisse.

Um einem bereits definierten zusammengesetzten Ereignis weitere Unterereignisse hinzuzufügen, verwenden Sie den Befehl ADD SUBEVENT. Es gibt keine Begrenzung für die Anzahl von Unterereignissen, die Sie mit dem Befehl ADD SUBEVENT hinzufügen können.

Anmerkung: Die folgenden Typen von Ereignissen können einem zusammengesetzten Ereignis *nicht* als Unterereignisse hinzugefügt werden:

- Zusammengesetzte Ereignisse
- Systemereignisse
- Unterereignisse anderer zusammengesetzter Ereignisse
- Eingabeereignisse, wenn das zusammengesetzte Ereignis den Operator AND verwendet

Zum Entfernen von Unterereignissen aus einem zusammengesetzten Ereignis verwenden Sie den Befehl REMOVE SUBEVENT.

Optionen

AND

Gibt an, dass dem Prädikat dieses zusammengesetzten Ereignisses der boolesche Operator AND zugeordnet werden soll. Dies bedeutet, dass das zusammengesetzte Ereignis ausgelöst wird, wenn *alle* zugehörigen Unterereignisse ausgelöst wurden.

Anmerkung: Der Ausführungsstatus eines leeren zusammengesetzten Ereignisses, das den Operator AND verwendet, ist immer FIRED (wahr).

EVENT(datenwert)

Gibt den Namen (1 - 16 Zeichen) des zusammengesetzten Ereignisses an, das definiert wird. Zulässige Zeichen: A-Z a-z 0-9 \$ @ # . - und _. Führende und eingebettete Leerzeichen sind nicht zulässig. Wenn der angegebene Name kürzer als 16 Zeichen ist, wird er mit nachfolgenden Leerzeichen bis auf 16 Zeichen aufgefüllt.

OR

Gibt an, dass dem Prädikat dieses zusammengesetzten Ereignisses der boolesche Operator OR zugeordnet werden soll. Dies bedeutet, dass das zusammengesetzte Ereignis ausgelöst wird, wenn ein *beliebiges* der zugehörigen Unterereignisse ausgelöst wird.

Anmerkung: Der Ausführungsstatus eines leeren zusammengesetzten Ereignisses, das den Operator OR verwendet, ist immer NOTFIRED (falsch).

SUBEVENTn(datenwert)

Gibt den Namen (1 - 16 Zeichen) eines Unterereignisses an, das dem zusammengesetzten Ereignis hinzugefügt werden soll, wenn das zusammengesetzte Ereignis erstellt wird. Zulässige Zeichen: A-Z a-z 0-9 \$ @ # . - und _. Führende und eingebettete Leerzeichen sind nicht zulässig. Wenn der angegebene Name kürzer als 16 Zeichen ist, wird er mit nachfolgenden Leerzeichen bis auf 16 Zeichen aufgefüllt.

Sie können diese Option bis zu achtmal angeben. Der Wert *n* muss im Bereich von 1 - 8 liegen.

Die Unterereignisse, die Sie angeben, müssen zuvor für die aktuelle Aktivität durch Befehle DEFINE INPUT EVENT, DEFINE ACTIVITY oder DEFINE TIMER definiert worden sein. Sie dürfen keine Unterereignisse vorhandener zusammengesetzter Ereignisse sein.

Bedingungen

111 EVENTERR

RESP2-Werte:

6

Der in der Option EVENT angegebene Ereignisname ist ungültig.

7

Der in der Option EVENT angegebene Ereignisname wurde bereits für diese Aktivität definiert.

21–28

Eines oder mehrere der Unterereignisse, die in der Option SUBEVENTn angegeben wurden, ist nicht vorhanden. Der RESP2-Wert gibt das erste Unterereignis an, das nicht vorhanden ist.

16 INVREQ

RESP2-Werte:

1

Der Befehl wurde außerhalb des Gültigkeitsbereichs einer Aktivität abgesetzt.

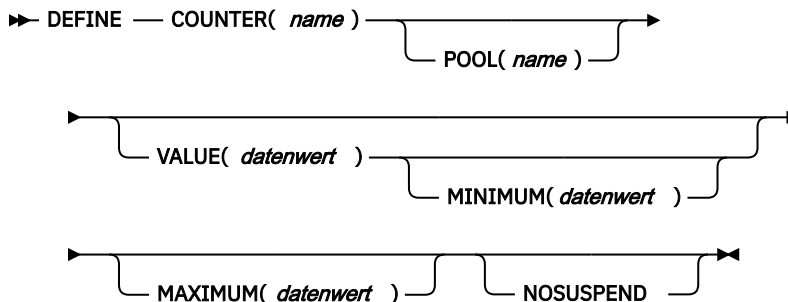
31–38

Einer oder mehrere der Unterereignisnamen, die in der Option SUBEVENTn angegeben wurden, ist ungültig. Der RESP2-Wert gibt den ersten ungültigen Unterereignisnamen an.

DEFINE COUNTER und DEFINE DCOUNTER

Erstellt einen benannten Zähler in einem benannten Zählerpool in der Coupling-Facility. Mit COUNTER werden Zähler erstellt, die als Vollwortzähler mit Vorzeichen betrachtet werden, mit DCOUNTER werden Zähler erstellt, die als Doppelwortzähler ohne Vorzeichen betrachtet werden.

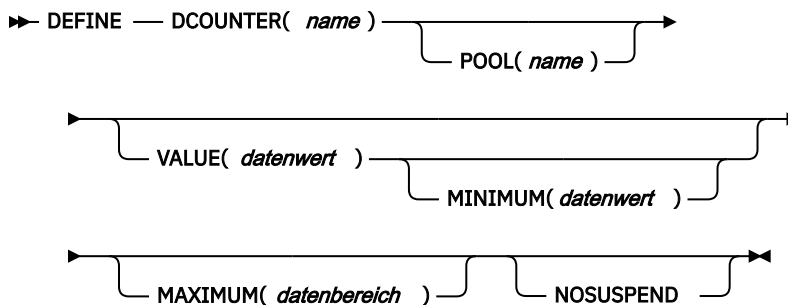
DEFINE COUNTER



Bedingungen: BUSY, INVREQ

Dieser Befehl ist threadsicher.

DEFINE DCOUNTER



Bedingungen: BUSY, INVREQ

Dieser Befehl ist threadsicher.

Beschreibung

Diese Zählerbefehle erstellen einen neuen benannten Zähler in einem benannten Zählerpool in der Coupling-Facility.

Obwohl Sie die CICS-API zum Arbeiten mit Vollwort-Binärwerten (mit Vorzeichen) oder Doppelwort-Binärwerten (ohne Vorzeichen) verwenden können, speichert der Server für benannte Zähler alle Werte als Doppelwortwerte ohne Vorzeichen. Es können Überlaufbedingungen auftreten, zum Beispiel wenn Sie einen Zähler mit dem Befehl DCOUNTER definieren und versuchen, auf diesen Zähler mit dem Befehl

COUNTER zuzugreifen. Greifen Sie auf einen benannten Zähler mit Befehlen aus demselben Befehlssatz zu, der auch zum Definieren des Zählers verwendet wurde.

Informationen zur Angabe von Vollwort- und Doppelwortvariablen bei diesen Befehlen für benannte Zähler finden Sie unter „Argumentwerte für CICS-Befehle“ auf Seite 2.

Optionen

COUNTER(name)

Gibt das 16 Byte lange Feld für den Zählernamen an. Der Name muss mit einem Buchstaben beginnen, kann Buchstaben, Ziffern oder Unterstreichungszeichen enthalten und muss mit nachgestellten Leerzeichen auf 16 Byte aufgefüllt werden. (Buchstaben müssen großgeschrieben werden. Die Symbole mit landesspezifischer Verwendung "£", "#" und "@" sind überall dort zulässig, wo Buchstaben zulässig sind.) Es wird empfohlen, dass der Name mit einem systematischen und für die Anwendung eindeutigen Präfix beginnen soll. Alle Zähler, die von CICS selbst verwendet werden, verwenden das Präfix "DFH".

DCOUNTER(name)

Gibt das 16 Byte lange Feld für den Zählernamen an. Der Name muss mit einem Buchstaben beginnen, kann Buchstaben, Ziffern oder Unterstreichungszeichen enthalten und muss mit nachgestellten Leerzeichen auf 16 Byte aufgefüllt werden. (Buchstaben müssen großgeschrieben werden. Die Symbole mit landesspezifischer Verwendung "£", "#" und "@" sind überall dort zulässig, wo Buchstaben zulässig sind.) Es wird empfohlen, dass der Name mit einem systematischen und für die Anwendung eindeutigen Präfix beginnen soll. Alle Zähler, die von CICS selbst verwendet werden, verwenden das Präfix "DFH".

MAXIMUM(datenwert)

Gibt die maximale Nummer für den benannten Zähler an, wobei ein Vollwort-Binärwert mit Vorzeichen für COUNTER und ein Doppelwort-Binärwert ohne Vorzeichen für DCOUNTER verwendet wird. Dies ist die höchste Nummer, die in einem Befehl GET zugeordnet werden kann. Anschließend muss der Zähler durch einen Befehl REWIND zurückgesetzt werden.

Wenn Sie den Parameter MAXIMUM nicht angeben, wird der benannte Zähler mit einem Standardmaximum hoher Werte (X'7FFFFFFF' im Fall des Vollwortwerts mit Vorzeichen oder ein mit X'FF' gefüllter Doppelwortwert) definiert.

MINIMUM(datenwert)

Gibt die minimale Nummer für den benannten Zähler an, wobei ein Vollwort-Binärwert mit Vorzeichen für COUNTER und ein Doppelwort-Binärwert ohne Vorzeichen für DCOUNTER verwendet wird. Dies ist der Wert, auf ein benannter Zähler durch einen Befehl REWIND zurückgesetzt wird.

Wenn Sie den Parameter MINIMUM angeben, müssen Sie auch einen Parameter VALUE angeben.

Wenn Sie den Parameter MINIMUM nicht angeben, wird der benannte Zähler mit einem Standardminimum niedriger Werte (mit X'00' gefülltes Voll- oder Doppelwort) definiert.

NOSUSPEND

Gibt eine sofortige Rückkehr zum Anwendungsprogramm während einer Neuerstellung der Coupling-Facility-Struktur an. Wenn Sie den Parameter NOSUSPEND nicht angeben, wird die Anforderung so lange wiederholt, bis sie erfolgreich ist. Wenn Sie NOSUSPEND angeben, schlägt die Anforderung sofort fehl und CICS gibt die Bedingung BUSY, RESP2=500 zurück.

POOL(name)

Gibt eine acht Zeichen lange Zeichenfolge an, die als Poolauswahlparameter verwendet werden soll, um den Pool auszuwählen, in dem der benannte Zähler erstellt werden soll. Die Zeichenfolge kann ein logischer Poolname oder der tatsächliche Poolname sein.

Gültige Zeichen für die Poolauswahlzeichenfolge: A bis Z, 0 bis 9, \$ @ # und _ (Unterstreichungszeichen). Wenn *name* eine Variable ist, die einen Namen enthält, der weniger als 8 Zeichen lang ist, muss der Name mit nachfolgenden Leerzeichen aufgefüllt werden.

Dieser Parameter ist optional. Wenn Sie den Namen des Pools nicht angeben, wird ein Poolauswahlwert aus 8 Leerzeichen angenommen.

Wenn kein entsprechender Eintrag in der Optionstabelle DFHNCOPT vorhanden ist, verwendet CICS den Namen, der im Systeminitialisierungsparameter NCPLDFT angegeben ist, der den Standardpool für benannte Zähler angibt.

Informationen zur Generierung einer Optionstabelle für benannte Zähler mit dem Makro DFHNCO finden Sie unter [Server für benannte Zähler](#).

VALUE(datenwert)

Gibt den Anfangswert an, bei dem der neue benannte Zähler beginnen soll, wobei ein Vollwort-Binärwert mit Vorzeichen für COUNTER und ein Doppelwort-Binärwert ohne Vorzeichen für DCOUNTER verwendet wird.

Sie können einen Wert angeben, der zwischen dem Minimalwert (einschließlich) und dem Maximalwert plus 1 liegt. Wenn Sie als Anfangswert den Maximalwert plus 1 angeben, wird der Zähler mit der Bedingung 'Zähler am Grenzwert' erstellt, sodass er erst verwendet werden kann, wenn er zurückgesetzt (REWIND) wurde.

Wenn Sie weder den Parameter VALUE noch den Parameter MINIMUM angeben, wird der Zähler mit dem Anfangswert null erstellt. Wenn Sie VALUE weglassen, jedoch MINIMUM angeben, gibt der Umsetzer einen Fehler aus. Der Parameter VALUE ist erforderlich, wenn Sie den Parameter MINIMUM angeben.

Bedingungen

128 BUSY

RESP2-Werte:

500

Die Option NOSUSPEND wurde im Befehl angegeben und die Coupling-Facility-Struktur ist momentan während einer Neuerstellung nicht verfügbar.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

RESP2-Werte:

202

Doppelter Zählername. Ein benannter Zähler dieses Namens ist bereits vorhanden.

301

Der Server hat einen Fehlercode gemeldet, der für die Schnittstelle des benannten Zählers unverständlich ist. Dies ist in der Regel nur dann möglich, wenn das Schnittstellenlademodul DFHNCF eine geringere Wartungsstufe oder ein niedrigeres Release-Level als der Server selbst aufweist.

302

Der Server kann den neuen benannten Zähler nicht erstellen, weil nicht genügend Platz in dem Pool für benannte Zähler vorhanden ist.

303

Bei einem Makro, das für den Zugriff auf die Coupling-Facility verwendet wird, trat ein unerwarteter Fehler wie beispielsweise eine Strukturstörung oder eine Verbindungsunterbrechung auf. Weitere Informationen enthält die Nachricht DFHNC0441 im Jobprotokoll der Anwendung.

304

Der im Programm angegebene Parameter für die Poolauswahl kann anhand der aktuellen Optionstabelle nicht in einen gültigen Servernamen aufgelöst werden.

305

Die Schnittstelle kann keine Verbindung zum Server für den ausgewählten Pool mit benannten Zählern herstellen. Weitere Informationen enthält eine AXM-Servicenachricht (AXMSCnnnn) im Jobprotokoll der Anwendung.

306

Während der Server eine Anforderung verarbeitete, trat eine abnormale Beendigung auf. Weitere Informationen enthält eine Nachricht im Jobprotokoll der Anwendung und im Jobprotokoll des Servers.

308

Das Optionstabellenmodul DFHNCOPT, das zum Auflösen eines Poolnamens benötigt wird, konnte nicht geladen werden.

309

Während der Verarbeitung der Optionstabelle hat die Schnittstelle für benannte Zähler ein unbekanntes Eintragsformat festgestellt. Entweder wurde die Optionstabelle nicht ordnungsgemäß generiert oder das Schnittstellenlademodul DFHNCIF hat nicht dasselbe Release-Level wie die Optionstabelle

310

Ein Optionstabelleneintrag, der mit dem angegebenen Poolnamen übereinstimmt, hat ein Benutzerexitprogramm angegeben, aber das Benutzerexitprogramm ist nicht durch eine Linkbearbeitung mit der Optionstabelle verbunden und kann nicht geladen werden.

311

Eine Antwort vom Server des benannten Zählers an das Schnittstellenmodul DFHNCIF der Client-region gibt an, dass gegenwärtig ein systemverwalteter erneuter Build ausgeführt wird, aber die Bedingung wird von der EXEC CICS-Schnittstelle nicht erkannt. Dies bedeutet, dass die CICS-Region den Stand von CICS TS 2.1 oder früher aufweist.

403

Der Parameter POOL enthält ungültige Zeichen oder eingebettete Leerzeichen.

404

Der Parameter COUNTER enthält ungültige Zeichen oder eingebettete Leerzeichen.

406

Der Parameter VALUE ist ungültig. Sie können den aktuellen Wert nicht kleiner als den Mindestwert oder größer als den Maximalwert plus 1 festlegen.

407

Der Parameter MINIMUM oder MAXIMUM ist ungültig. Entweder gibt der Parameter MAXIMUM einen Wert an, der kleiner als der Minimalwert ist oder (nur für COUNTER) einer der Parameter gibt einen negativen Wert an.

Standardaktion: Die Task wird abnormal beendet.

DEFINE INPUT EVENT

Definiert ein BTS-Eingabeereignis.

DEFINE EVENT

➤ DEFINE — INPUT — EVENT(*datenwert*) ➤

Bedingungen: EVENTERR, INVREQ

Beschreibung

Der Befehl DEFINE INPUT EVENT definiert ein Eingabeereignis für Business Transaction Services (BTS). Ein Eingabeereignis wird in der Regel durch eine übergeordnete Aktivität an eine Aktivität übergeben, wodurch die Aktivität aktiviert wird. (Manchmal kommt das Eingabeereignis jedoch von außerhalb des Prozesses.)

Die meisten Ereignisse werden bei Beendigung eines Vorgangs, wie zum Beispiel einer Aktivität oder eines angegebenen Zeitintervalls, ausgelöst. Ein Eingabeereignis ist insofern anders, als dass es ausgelöst wird, nachdem ein Befehl RUN abgesetzt wurde, der das Ereignis angibt.

Eine Aktivität definiert ein Eingabeereignis, um eine Benachrichtigung (durch die Option INPUTEVENT des Befehls RUN oder LINK ACTIVITY) über den Grund zu empfangen, aus dem sie aktiviert wurde.

Anmerkung: Systemereignisse wie DFHINITIAL sind ein spezieller Typ von Eingabeereignis. Sie werden von allen Aktivitäten erkannt und müssen nicht definiert werden.

Optionen

EVENT(datenwert)

Gibt den Namen (1 - 16 Zeichen) des Eingabeereignisses an, das definiert wird. Zulässige Zeichen: A-Z a-z 0-9 \$ @ # . - und _. Führende und eingebettete Leerzeichen sind nicht zulässig. Wenn der angegebene Name kürzer als 16 Zeichen ist, wird er mit nachfolgenden Leerzeichen bis auf 16 Zeichen aufgefüllt.

Bedingungen

111 EVENTERR

RESP2-Werte:

6

Der in der Option EVENT angegebene Ereignisname ist ungültig.

7

Der in der Option EVENT angegebene Ereignisname wurde bereits für diese Aktivität definiert.

16 INVREQ

RESP2-Werte:

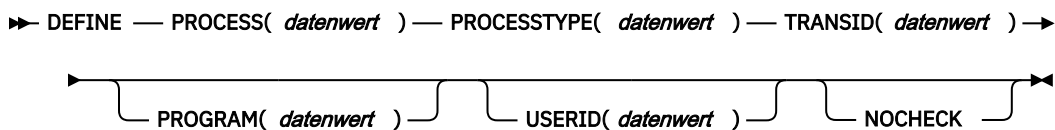
1

Der Befehl wurde außerhalb des Gültigkeitsbereichs einer Aktivität abgesetzt.

DEFINE PROCESS

Definiert einen CICS-BTS-Prozess.

DEFINE PROCESS



Bedingungen: INVREQ, IOERR, NOTAUTH, PROCESSERR, TRANSIDERR

Beschreibung

Der Befehl DEFINE PROCESS definiert einen BTS-Prozess. Er führt die folgenden Aktionen aus:

- Er fügt einen neuen Prozess (z. B. eine neue Instanz einer Geschäftstransaktion) dem CICS-BTS-System hinzu.
- Er erstellt die Stammaktivität des Prozesses.

Der Name des Programms, das in der Ausführung des neuen Prozesses verwendet wird, wird entweder der Option PROGRAM entnommen oder, falls PROGRAM nicht angegeben wird, der Transaktionsdefinition entnommen, auf die die Option TRANSID verweist.

Die Transaktionsattribute, die in den Optionen TRANSID und USERID angegeben werden, werden wirksam, wenn der Prozess durch einen Befehl RUN aktiviert wird, jedoch *nicht*, wenn er durch einen Befehl LINK aktiviert wird (siehe „RUN“ auf Seite 476).

BTS schreibt das Hinzufügen des Prozesses erst fest, wenn die anfordernde Transaktion einen erfolgreichen Synchronisationspunkt erfasst hat.

Optionen

NOCHECK

Gibt an, dass kein Datensatz in die Repository-Datei geschrieben werden soll, um den Namen des Prozesses zu reservieren.

Beachten Sie, dass der Prozessname in dem Repository eindeutig sein muss (siehe Optionen PROCESS und PROCESSTYPE) und dass BTS das Hinzufügen des Prozesses erst festschreibt, wenn die anfordernde Transaktion einen erfolgreichen Synchronisationspunkt erfasst hat.

Sie können diese Option verwenden, um die BTS-Leistung zu verbessern, indem Sie den Schreibvorgang in das Repository und die damit verbundene Protokollierung unterdrücken. In diesem Fall müssen Sie allerdings beachten, dass ein Fehler durch Angabe eines nicht eindeutigen Prozessnamens nicht mehr zur Folge hat, dass die Bedingung PROCESSERR für den Befehl DEFINE PROCESS zurückgegeben wird. Der Fehler wird möglicherweise erst viel später entdeckt, wenn ein Synchronisationspunkt erfolgt, sodass die Fehlerbehebung wesentlich schwieriger wird.

PROCESS(datenwert)

Gibt einen Namen (1-36 Zeichen) an, um den neuen Prozess (Geschäftstransaktionsinstanz) zu identifizieren. Der Name muss innerhalb der BTS-Repository-Datei eindeutig sein, in der Details des Prozesses gespeichert werden sollen (siehe Option PROCESSTYPE). Es ist zum Beispiel gültig, einen Befehl DEFINE abzusetzen, in dem die Option PROCESS einen Namen angibt, der zurzeit von einem anderen Prozess verwendet wird, *vorausgesetzt* die Option PROCESSTYPE verweist auf eine andere zugrunde liegende Repository-Datei als die, in der der erste Prozess definiert ist.

Zulässige Zeichen: A-Z a-z 0-9 \$ @ # / % & ? ! : | " = ~ , ; < > . - und _ . Führende und eingebettete Leerzeichen sind ebenfalls zulässig.

Wenn der Name als Literalzeichenfolge angegeben wird, die kürzer als 36 Zeichen ist, wird er mit nachfolgenden Leerzeichen bis auf 36 Zeichen aufgefüllt. Wenn der Name als Variable angegeben wird, deren Wert weniger als 36 Zeichen lang ist, erfolgt keine Auffüllung.

PROCESSTYPE(datenwert)

Gibt den Typ (1-8 Zeichen) des neuen Prozesses an.

Jeder Prozesstyp verweist auf eine VSAM-Datei (das Repository), in der Informationen zu Prozessen des angegebenen Typs gespeichert werden. Das heißt, Informationen zum Status eines Prozesses (und seiner zugehörigen Teilaktivitäten) werden in dem Repository gespeichert, das dem Prozesstyp zugeordnet ist, zu dem der Prozess gehört. Datensätze für mehrere Prozesstypen können in derselben Repository-Datei gespeichert werden.

Sie können Ihre Prozesse kategorisieren, indem Sie ihnen verschiedene Prozesstypen zuweisen.

PROGRAM(datenwert)

Gibt den Namen (1-8 Zeichen) des Programms für den Prozess an, der definiert wird. Wenn kein Programm angegeben wird, wird der Name aus der TRANSID-Definition genommen.

TRANSID(datenwert)

Gibt den Namen (1-4 Zeichen) der Transaktion an, unter der der Prozess ausgeführt werden soll, wenn er durch einen Befehl RUN aktiviert wird.

Anmerkung: Wenn der Prozess durch einen Befehl LINK aktiviert wird, wird er unter der Transaktions-ID (TRANSID) der Transaktion ausgeführt, die den Befehl LINK absetzt.

Die Transaktion muss in der CICS-Region definiert sein, in der der Befehl DEFINE PROCESS ausgeführt wird.

USERID(datenwert)

Gibt die Benutzer-ID (1-8 Zeichen) an, unter deren Berechtigung der Prozess ausgeführt werden soll, wenn er durch einen Befehl RUN aktiviert wird.

Anmerkung: Wenn der Prozess durch einen Befehl LINK aktiviert wird, wird er unter der Benutzer-ID (USERID) der Transaktion ausgeführt, die den Befehl LINK absetzt.

Der Wert dieses Felds wird als *definierte Benutzer-ID* bezeichnet.

Wenn Sie die Option USERID nicht angeben, wird als definierte Benutzer-ID standardmäßig die Benutzer-ID verwendet, unter der die Transaktion, die den Befehl DEFINE absetzt, ausgeführt wird. Diese Benutzer-ID wird als *Befehlsbenutzer-ID* bezeichnet.

Wird die Option USERID angegeben, führt CICS (zum Zeitpunkt der Definition) eine Ersatzsicherheitsprüfung durch, um sicherzustellen, dass die Befehlsbenutzer-ID berechtigt ist, die definierte Benut-

zer-ID zu verwenden. Wenn Sie also die Option USERID angeben, müssen Sie die Befehlsbenutzer-ID als Ersatzbenutzer der definierten Benutzer-ID berechtigen.

Bedingungen

16 INVREQ

RESP2-Werte:

12

Der installierte Prozesstyp (PROCESSTYPE) ist nicht aktiviert.

22

Die Arbeitseinheit, die den Befehl DEFINE PROCESS abgesetzt hat, hat bereits eine Aktivität angefordert.

17 IOERR

RESP2-Werte:

29

Die Repository-Datei ist nicht verfügbar.

30

Es ist ein Ein-/Ausgabefehler für die Repository-Datei aufgetreten.

70 NOTAUTH

RESP2-Werte:

101

Der Benutzer, der der absetzenden Task zugeordnet ist, ist nicht berechtigt, auf die Datei zuzugreifen, die der BTS-Repository-Datei zugeordnet ist, in der Details des Prozesses gespeichert werden sollen.

102

Der Benutzer, der der absetzenden Task zugeordnet ist, ist nicht als Ersatz für die definierte Benutzer-ID berechtigt, die in der Option USERID angegeben wurde.

108 PROCESSERR

RESP2-Werte:

2

Der Prozessname, der in der Option PROCESS angegeben wurde, wird in der BTS-Repository-Datei bereits verwendet, die der Option PROCESSTYPE zugeordnet ist.

9

Der in der Option PROCESSTYPE angegebene Prozesstyp konnte nicht gefunden werden.

16

Der Prozessname, der in der Option PROCESS angegeben wurde, enthält ein oder mehrere ungültige Zeichen.

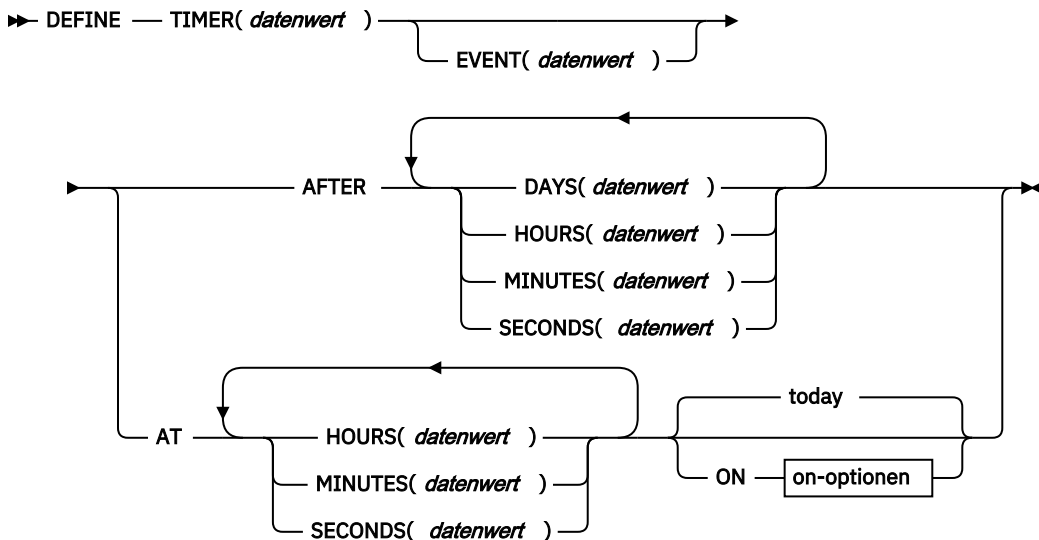
28 TRANSIDERR

Die in der Option TRANSID angegebene Transaktions-ID ist für CICS nicht definiert.

DEFINE TIMER

Definiert einen BTS-Zeitgeber.

DEFINE TIMER



on-optionen



Bedingung: EVENTERR, INVREQ, TIMERERR

Beschreibung

Der Befehl **DEFINE TIMER** definiert einen BTS-Zeitgeber, der nach dem angegebenen Intervall oder zu einem angegebenen Zeitpunkt (Datum und Uhrzeit) abläuft. Wenn ein Zeitgeber definiert wird, wird auch ein zugeordnetes Ereignis im Ereignispool der aktuellen Aktivität definiert. Als Name des zugeordneten Ereignisses wird standardmäßig der Name des Zeitgeber angenommen. Wenn der Zeitgeber abläuft, wird das zugeordnete Ereignis ausgelöst.

Anmerkung:

1. Alle Datums- und Zeitangaben beziehen sich auf die Ortszeit.
2. Ein Zeitgeber, der einen Zeitpunkt (Datum und Uhrzeit) angibt, der bereits vergangen ist, läuft sofort ab. Ebenso läuft der Zeitgeber sofort ab, wenn das angeforderte Intervall null ist.

Optionen

AFTER

Gibt ein Zeitintervall an, das abgewartet werden soll, bevor der Zeitgeber ablaufen soll.

Sie müssen mindestens eine der Optionen DAYS(0–999), HOURS(0–23), MINUTES(0–59) und SECONDS(0–59) angeben. Beispiel: Die Angabe HOURS(1) SECONDS(3) bedeutet 1 Stunde und 3 Sekunden (für Minuten wird der Standardwert 0 angenommen).

AT

Gibt den Zeitpunkt an, zu dem der Zeitgeber ablaufen soll.

Sie müssen mindestens eine der Optionen HOURS(0–23), MINUTES(0–59) und SECONDS(0–59) angeben. Beispiel:

- Die Angabe HOURS(1) bedeutet 01:00 Uhr.
- Die Angabe HOURS(15) MINUTES(15) bedeutet 15:15 Uhr.
- Die Angabe MINUTES(15) bedeutet 0:15 Uhr.

DAYOFMONTH(datenwert)

Gibt den Tag des Monats, an dem der Zeitgeber ablaufen soll, als Vollwort-Binärwert aus dem Bereich von 1 - 31 an.

DAYOFYEAR(datenwert)

Gibt den Tag des Jahres, an dem der Zeitgeber ablaufen soll, als Vollwort-Binärwert aus dem Bereich von 1 - 366 an. Zum Beispiel gibt DAYOFYEAR(1) den 1. Januar an.

DAYS(datenwert)

Gibt einen Vollwort-Binärwert aus dem Bereich von 0 - 999 an. Dies ist eine Unteroption der Option AFTER. Informationen zur Verwendung und zur Bedeutung finden Sie unter der Option AFTER.

Der Standardwert ist null.

EVENT(datenwert)

Gibt den Namen (1 - 16 Zeichen) des Ereignisses an, das dem Zeitgeber zugeordnet werden soll. Zulässige Zeichen: A-Z a-z 0-9 \$ @ # . - und _. Führende und eingebettete Leerzeichen sind nicht zulässig. Wenn der angegebene Name kürzer als 16 Zeichen ist, wird er mit nachfolgenden Leerzeichen bis auf 16 Zeichen aufgefüllt.

Der Standardereignisname ist der Name des Zeitgebers.

HOURS(datenwert)

Gibt einen Vollwort-Binärwert aus dem Bereich von 0 - 23 an. Dies ist eine Unteroption der Optionen AFTER und AT. Informationen zur Verwendung und zur Bedeutung finden Sie unter diesen Optionen.

Der Standardwert ist null.

MINUTES(datenwert)

Gibt einen Vollwort-Binärwert aus dem Bereich von 0 - 59 an. Dies ist eine Unteroption der Optionen AFTER und AT. Informationen zur Verwendung und zur Bedeutung finden Sie unter diesen Optionen.

Der Standardwert ist null.

MONTH(datenwert)

Gibt den Monat, in dem der Zeitgeber ablaufen soll, als Vollwort-Binärwert aus dem Bereich von 1 - 12 an.

ON

Gibt das Datum, an dem der Zeitgeber ablaufen soll, als Kombination aus den Optionen YEAR, MONTH, DAYOFMONTH und DAYOFYEAR an.

Wenn die Option ON nicht angegeben wird, wird standardmäßig das heutige Datum angenommen.

SECONDS(datenwert)

Gibt einen Vollwort-Binärwert aus dem Bereich von 0 - 59 an. Dies ist eine Unteroption der Optionen AFTER und AT. Informationen zur Verwendung und zur Bedeutung finden Sie unter diesen Optionen.

Der Standardwert ist null.

TIMER(datenwert)

Gibt den Namen (1 - 16 Zeichen) des Zeitgebers an. Zulässige Zeichen: A-Z a-z 0-9 \$ @ # . - und _. Führende und eingebettete Leerzeichen sind nicht zulässig. Wenn der angegebene Name kürzer als 16 Zeichen ist, wird er mit nachfolgenden Leerzeichen bis auf 16 Zeichen aufgefüllt.

YEAR(datenwert)

Gibt das Jahr, in dem der Zeitgeber ablaufen soll, als Vollwort-Binärwert aus dem Bereich von 0 - 2040 an.

Bedingungen**111 EVENTERR**

RESP2-Werte:

6

Der in der Option EVENT angegebene Ereignisname ist ungültig.

7

Der in der Option EVENT angegebene Ereignisname (oder der dem Zeitgebernamen entnommene Standardereignisname) wurde für diese Aktivität bereits definiert.

16 INVREQ

RESP2-Werte:

1

Der Befehl wurde außerhalb des Gültigkeitsbereichs einer momentan aktiven Aktivität abgesetzt.

11

Es wurde ein ungültiges Intervall angegeben.

12

Es wurde ein ungültiges Datum oder eine ungültige Uhrzeit angegeben.

115 TIMERERR

RESP2-Werte:

14

Der in der Option TIMER angegebene Zeitgeber ist ungültig.

15

Der in der Option TIMER angegebene Zeitgeber wurde bereits für diese Aktivität definiert.

Beispiele

```
DEFINE TIMER() AT HOURS(15)
```

Dieser Befehl definiert einen Zeitgeber, der heute um 03:00 Uhr (oder sofort, falls die Ortszeit bereits nach 03:00 Uhr liegt) abläuft.

```
DEFINE TIMER() AT HOURS(15) ON YEAR(2001) MONTH(11) DAYOFMONTH(3)
```

Dieser Befehl definiert einen Zeitgeber, der um 15:00 Uhr am 3. November 2001 abläuft.

```
DEFINE TIMER() AT HOURS(15) ON YEAR(2001) DAYOFYEAR(32)
```

Dieser Befehl definiert einen Zeitgeber, der um 15:00 Uhr am 1. Februar 2001 abläuft.

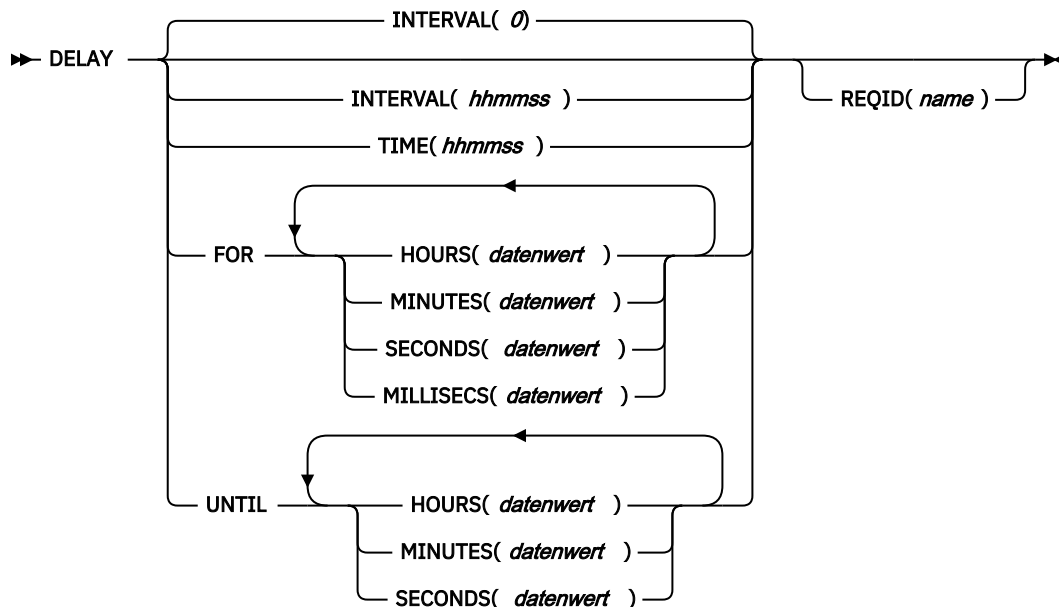
```
DEFINE TIMER() AT HOURS(8) ON YEAR(1997) MONTH(1) DAYOFMONTH(1)
```

Dieser Befehl definiert einen Zeitgeber, der sofort abläuft.

DELAY

Verzögert die Verarbeitung einer Task.

DELAY



Bedingungen: EXPIRED, INVREQ, NORMAL

Dieser Befehl ist nur threadsicher, wenn das Intervall 0 ist.

Hinweis für dynamisches Transaktionsrouting: Die Verwendung des Befehls DELAY mit der Option REQID kann bei späterem Abbruch durch CANCEL möglicherweise transaktionsübergreifende Affinitäten erstellen, die sich negativ auf die Verwendung des dynamischen Transaktionsroutings auswirken. Weitere Informationen zu Transaktionsaffinitäten finden Sie unter [Affinität](#).

Beschreibung

Der Befehl DELAY setzt die Verarbeitung der absetzenden Task für ein angegebenes Zeitintervall oder bis zu einer angegebenen Tageszeit aus. Er ersetzt jeden zuvor initialisierten Befehl POST für die Task.

Es ist möglich, Intervalle in Millisekunden (ms) anzugeben. CICS prüft jedoch alle 250 ms auf den Ablauf von Verzögerungsintervallen. Das heißt, das tatsächliche Intervall kann abhängig davon variieren, an welcher Position im Prüfzyklus die Anforderung erfolgt.

Das Standardintervall ist INTERVAL(0); in der Sprache C ist die Standardangabe jedoch FOR HOURS(0) MINUTES(0) SECONDS(0).

Optionen

FOR

Gibt die Dauer der Verzögerung an.

HOURS(datenwert)

Gibt einen Vollwort-Binärwert im Bereich von 0 bis 99 an.

INTERVAL(hhmmss)

Gibt ein Zeitintervall im gepackten Dezimalformat an, das ab dem Zeitpunkt, zu dem der Befehl DELAY abgesetzt wird, ablaufen soll. Die Werte **mm** und **ss** liegen im Bereich von 0 bis 59. Die angegebene Zeit wird von CICS der aktuellen Systemzeit hinzuaddiert, wenn der Befehl ausgeführt wird, um die Ablaufzeit zu berechnen.

Bei Verwendung der Programmiersprache C wird empfohlen, die Optionen FOR/UNTIL HOURS, MINUTES und SECONDS zu verwenden, da C keinen gepackten Dezimaldatentyp bereitstellt. Sie können INTERVAL verwenden, wenn jedoch der angegebene Wert **keine** ganzzahlige Konstante ist, obliegt es der Anwendung, sicherzustellen, dass der an CICS übergebene Wert das gepackte Dezimalformat hat.

MILLISECS(datenwert)

Gibt einen Vollwort-Binärwert im Bereich von 0 bis 999 an, wenn auch die Optionen HOURS, MINUTES oder SECONDS angegeben werden, oder gibt einen Wert im Bereich von 0 bis 359999999 an, wenn MILLISECS die einzige Option ist, die angegeben wird.

Anmerkung: CICS behandelt Verzögerungszeiträume von weniger als 50 Millisekunden als ob die Zeit sofort abgelaufen ist.

MINUTES(datenwert)

Gibt einen Vollwort-Binärwert im Bereich von 0 bis 59 an, wenn auch die Optionen HOURS, SECONDS oder MILLISECS angegeben werden, oder gibt einen Wert im Bereich von 0 bis 5999 an, wenn MINUTES die einzige Option ist, die angegeben wird.

REQID(name)

Gibt einen Namen (1-8 Zeichen) an, der eindeutig sein muss, um die DELAY-Anforderung zu identifizieren.

Verwenden Sie diese Option zur Angabe eines anwendungsdefinierten Namens, um es einer anderen Transaktion zu ermöglichen, die DELAY-Anforderung abzuberechnen.

Wenn andere Tasks noch nicht abgelaufene DELAY-Anforderungen abbrechen können sollen, müssen Sie die Anforderungs-ID dynamisch verfügbar machen. Das Speichern einer Anforderungs-ID in einer TS-Warteschlange, deren Name anderen Anwendungen bekannt ist, die die DELAY-Anforderung möglicherweise abbrechen sollen, ist eine Methode, mit der Sie eine Anforderungs-ID an andere Transaktionen übergeben können.

Wenn Sie nach einer normalen Beendigung der Task ermitteln wollen, ob die DELAY-Anforderung normal abgelaufen ist oder durch einen Befehl **EXEC CICS CANCEL REQID** abgebrochen wurde, müssen Sie das Feld EIBRESP2 auf den Wert 23 testen, wenn die Steuerung zu der Anwendung zurückkehrt.

SECONDS(datenwert)

Gibt einen Vollwort-Binärwert im Bereich von 0 bis 59 an, wenn auch die Optionen HOURS, MINUTES oder MILLISECS angegeben werden, oder gibt einen Wert im Bereich von 0 bis 359 999 an, wenn SECONDS die einzige Option ist, die angegeben wird.

TIME(hhmmss)

Gibt einen Zeitpunkt im gepackten Dezimalformat an, zu dem die Task die Verarbeitung fortsetzen soll.

Bei Verwendung der Programmiersprache C wird empfohlen, die Optionen FOR/UNTIL HOURS, MINUTES und SECONDS zu verwenden, da C keinen gepackten Dezimaldatentyp bereitstellt. Sie können TIME verwenden, wenn jedoch der angegebene Wert **keine** ganzzahlige Konstante ist, obliegt es der Anwendung, sicherzustellen, dass der an CICS übergebene Wert das gepackte Dezimalformat hat. Lesen Sie in diesem Zusammenhang auch den Abschnitt über Ablaufzeiten unter [Intervallsteuerung](#).

UNTIL

Gibt den Zeitpunkt am Ende der Verzögerung an, zu dem die Task die Verarbeitung fortsetzen soll.

Bedingungen**0 NORMAL**

RESP2-Werte:

23

Gibt an, dass die DELAY-Anforderung durch eine andere Task abgebrochen wurde, die einen Befehl **CANCEL REQID** mit Angabe der eindeutigen ID abgesetzt hat, die von dieser Task verwendet wird.

31 EXPIRED

Tritt auf, wenn die angegebene Zeit bereits abgelaufen ist, wenn der Befehl abgesetzt wird. Wenn Sie sehr kurze Intervalle von weniger als 250 ms angeben, ist es wahrscheinlich, dass die Zeit abgelaufen ist, bevor der Befehl abgesetzt wird. Sie müssen eine Behandlung von Ausnahmerebedingungen vorsehen, wenn kurze Verzögerungsintervalle erforderlich sind.

Standardaktion: Die Bedingung wird ignoriert.

16 INVREQ

RESP2-Werte:

4

Stundenangabe außerhalb des gültigen Bereichs.

5

Minutenangabe außerhalb des gültigen Bereichs.

6

Sekundenangabe außerhalb des gültigen Bereichs.

22

Millisekundenangabe außerhalb des gültigen Bereichs.

Tritt außerdem (RESP2 nicht gesetzt) auf, wenn der Befehl DELAY für die Verarbeitung durch CICS nicht gültig ist.

Standardaktion: Die Task wird abnormal beendet.

Beispiele

Das folgende Beispiel zeigt, wie die Verarbeitung einer Task für fünf Minuten ausgesetzt wird:

```
EXEC CICS DELAY  
      INTERVAL(500)  
      REQID('GXLBZQMR')
```

```
EXEC CICS DELAY FOR MINUTES(5)
```

Das folgende Beispiel zeigt, wie um 09:00 Uhr die Verarbeitung einer Task bis 12:45 Uhr ausgesetzt wird:

```
EXEC CICS DELAY  
      TIME(124500)  
      REQID('UNIQUECODE')
```

Es gibt zwei Möglichkeiten, die Zeit unter UNTIL einzugeben.

- Eine Kombination aus mindestens zwei der Angaben HOURS(0-99), MINUTES(0-59) und SECONDS(0-59). Zum Beispiel würde die Angabe HOURS(1) SECONDS(3) eine Stunde und drei Sekunden bedeuten (für die Minuten wird standardmäßig null angenommen).
- Eine Angabe nur eines der Werte von HOURS(0-99), MINUTES(0-5999) oder SECONDS(0-359 999). Die Angabe HOURS(1) bedeutet 1 Stunde. Die Angabe MINUTES(62) bedeutet 1 Stunde und 2 Minuten. Die Angabe SECONDS(3723) bedeutet 1 Stunde, 2 Minuten und 3 Sekunden.

Unter FOR können Sie die Zeit auf beide Arten eingeben, die für UNTIL gelten, und Sie können zusätzlich eine Verzögerung angeben, um Millisekunden einzuschließen oder die Verzögerung insgesamt in Millisekunden anzugeben. Zur Angabe von Bruchsekunden für eine Verzögerung codieren Sie den Parameter MILLISECS im Bereich von 0 bis 999 zusätzlich zu anderen Zeiteinheiten. Zur Angabe der Verzögerung nur in Millisekunden codieren Sie den Parameter MILLISECS im Bereich von 0 bis 359999999. Das folgende Beispiel zeigt, wie Sie die Verarbeitung einer Task für 15000 Millisekunden aussetzen:

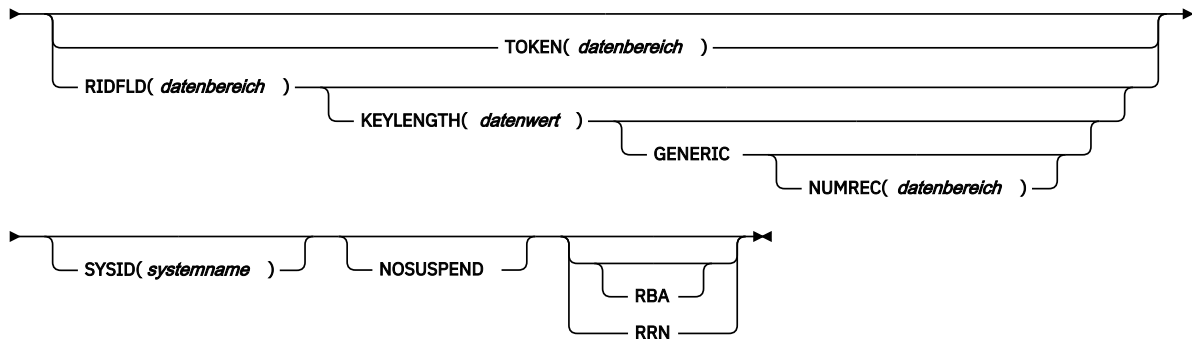
```
EXEC CICS DELAY  
      FOR MILLISECS(15000)  
      REQID('UNIQUECODE')
```

DELETE

Löscht einen Datensatz aus einer Datei - nur VSAM KSDS, VSAM RRDS und Datentabellen.

DELETE

→ DELETE — FILE(*dateiname*) →



Bedingungen: CHANGED, DISABLED, DUPKEY, FILENOTFOUND, ILLOGIC, INVREQ, IOERR, ISCVREQ, LOADING, LOCKED, NOTAUTH, NOTFND, NOTOPEN, RECORDBUSY, SYSIDERR

Dieser Befehl ist threadsicher, wenn für die Datei, auf die er sich bezieht, Folgendes zutrifft:

- Sie ist als fern definiert und der Befehl wird über eine IPIC-Verbindung durch Funktionsverlagerung an eine ferne CICS-Region übertragen.
- Sie ist als lokale VSAM-Datei, als VSAM-RLS-Datei oder als Coupling-Facility-Datentabelle definiert.

Dieser Befehl ist nicht threadsicher, wenn für die Datei, auf die er sich bezieht, Folgendes zutrifft:

- Sie ist als fern definiert und der Befehl wird über eine Nicht-IPIC-Verbindung durch Funktionsverlagerung übertragen.
- Sie ist als gemeinsam genutzte Datentabelle oder als BDAM-Datei definiert.

Beschreibung

Der Befehl **DELETE** löscht einen Datensatz aus einer Datei in einer KSDS, einen Pfad über eine KSDS, eine CICS- oder benutzerverwaltete Datentabelle oder einer RRDS. Sie können nicht aus einer VSAM ESDS- oder einer BDAM-Datei löschen. **Alle Verweise auf KSDS gelten ebenso für CICS-verwaltete Datentabellen und, sofern nicht anders angegeben, für Pfade über eine KSDS-Datei.** Die Datei kann sich auf einem lokalen oder einem fernen System befinden. Geben Sie in der Option RIDFLD den bestimmten Datensatz an, der gelöscht werden soll.

Sie können eine Gruppe von Datensätzen in einem einzelnen Aufruf dieses Befehls löschen, indem Sie die Gruppe durch die Option GENERIC (für RRDS nicht verfügbar) angeben. Der Befehl **DELETE** mit der Option GENERIC muss mit Vorsicht verwendet werden. Wenn der angegebene generische Schlüssel einer sehr hohen Anzahl von Datensätzen entspricht, fordert die CICS-Arbeitseinheit eine sehr hohe Anzahl von Datensatzsperrern an und behält diese bis zum Synchronisationspunkt bei. Dies könnte zu Speicherproblemen in der CICS-Region führen oder, bei einer RLS-Datei, eine übermäßige Nutzung von Coupling-Facility-Ressourcen verursachen und sich auf andere Systeme auswirken. Ziehen Sie stattdessen die Verwendung einer Reihe generischer Löschungen in Betracht (durch Verwendung generischer Schlüssel, die weniger Datensätzen entsprechen) und führen Sie Synchronisationspunkte nach jedem Bereich von gelöschten Datensätzen durch.

Sie können diesen Befehl auch verwenden, um einen einzelnen Datensatz zu löschen, der zuvor (durch einen Befehl **READ UPDATE**) zur Aktualisierung abgerufen wurde. In diesem Fall müssen Sie die RIDFLD nicht angeben.

Anmerkung: Wenn Sie eine Konvertierung in den RLS-Zugriffsmodus durchführen und den Systeminitialisierungsparameter **CILOCK=NO** in CICS verwenden, finden Sie Informationen und Hinweise zur Migration unter [VSAM RLS](#).

Wenn dieser Befehl zum Löschen von Datensätzen aus einer von CICS verwalteten Datentabelle verwendet wird, erfolgt die Aktualisierung sowohl an der VSAM-KSDS-Quellendatei (Dataset) als auch an der speicherinternen Datentabelle.

Wenn dieser Befehl zum Löschen von Datensätzen aus einer benutzerverwalteten Datentabelle verwendet wird, erfolgt die Aktualisierung nur an der speicherinternen Datentabelle.

Wenn dieser Befehl zum Löschen von Datensätzen aus einer Coupling-Facility-Datentabelle verwendet wird, erfolgt die Aktualisierung nur an der Datentabelle in der Coupling-Facility.

Optionen

FILE(dateiname)

Gibt den Namen der Datei an, auf die zugegriffen werden soll.

Wenn SYSID angegeben wird, wird angenommen, dass sich die Datei (Dataset), auf die sich diese Datei bezieht, auf einem fernen System befindet, und zwar unabhängig davon, ob der Name in CICS definiert ist. Ansonsten wird die Ressourcendefinition verwendet, um zu ermitteln, ob sich die Datei (Dataset) auf einem lokalen oder fernen System befindet.

GENERIC (nur_vsam_ksds)

Gibt an, dass der Suchschlüssel ein generischer Schlüssel ist, dessen Länge in der Option KEYLENGTH angegeben wird. Die Suche nach einem Datensatz wird erfüllt, wenn ein Datensatz mit einem Schlüssel gefunden wird, der dieselben Anfangszeichen (generischer Schlüssel) wie die angegebenen hat.

KEYLENGTH(datenwert)

Gibt die Länge (Halbwort-Binärwert) des Schlüssels an, der in der Option RIDFLD angegeben wurde, außer bei Angabe der Optionen RBA oder RRN, bei denen diese Option nicht gültig ist. Diese Option muss bei Angabe von GENERIC angegeben werden; sie kann immer dann angegeben werden, wenn ein Schlüssel angegeben ist. Wenn jedoch die Länge, die angegeben wird, von der Länge abweicht, die für die Datei (Dataset) definiert ist, und die Operation nicht generisch ist, tritt die Bedingung INVREQ auf.

Die Bedingung INVREQ tritt auch dann auf, wenn Sie GENERIC angeben und der Wert von KEYLENGTH nicht kleiner als die Länge ist, die in der VSAM-Definition angegeben ist.

Geben Sie für KEYLENGTH keinen Wert null an, da die Ergebnisse unvorhersehbar sind.

Für ferne Dateien können Sie die Option KEYLENGTH in der FILE-Definition angeben. Wenn KEYLENGTH nicht dort definiert wird und im Anwendungsprogramm nicht angegeben wird und der Schlüssel länger als 4 Zeichen ist, ist der Standardwert 4.

NOSUSPEND (nur RLS)

Gibt an, dass die Anforderung nicht warten soll, wenn VSAM eine aktive Sperre für den Datensatz hält. Dies gilt auch für Datensätze, die infolge eines Deadlocks gesperrt wurden.

Anmerkung: Anforderungen mit der Angabe NOSUSPEND warten mindestens eine Sekunde, bevor CICS die Antwort RECORDBUSY zurückgibt.

NUMREC(datenbereich) (Nur VSAM KSDS)

Gibt einen Halbwort-Binärwert-Datenbereich an, den CICS auf die Anzahl der gelöschten Datensätze setzt.

RBA

(Nur VSAM KSDS-Basisdateien, keine Pfade) Gibt an, dass das Datensatz-ID-Feld, das in der Option RIDFLD angegeben wurde, eine relative Byteadresse (RBA) enthält. Verwenden Sie diese Option nur, wenn Sie Datensätze unter Verwendung relativer Byteadressen anstelle von Schlüsseln zur Identifizierung der Datensätze löschen wollen.

Die Option RBA kann für Folgendes nicht verwendet werden:

- Benutzerverwaltete Datentabellen
- Coupling-Facility-Datentabellen
- Alle Dateien, die im RLS-Zugriffsmodus geöffnet sind

- KSDS-Dateien, die mehr als 4 GB Daten aufnehmen können

RIDFLD(datenbereich)

Gibt das Datensatz-ID-Feld an. Der Inhalt kann ein Schlüssel, eine relative Byteadresse (RBA) oder eine relative Satznummer (RRN) sein. Für eine relative Byteadresse oder eine relative Satznummer muss das Format dieses Felds ein Vollwort-Binärwert sein. Für eine relative Byteadresse kann der Wert von RIDFLD größer oder gleich null sein. Für eine relative Satznummer kann der Wert von RIDFLD größer oder gleich 1 sein.

Der Inhalt muss ein Schlüssel für benutzerverwaltete Datentabellen oder Coupling-Facility-Datentabellen sein.

Sie müssen diese Option angeben, wenn Sie auch GENERIC angeben.

RRN (nur VSAM RRDS)

Gibt an, dass das Datensatz-ID-Feld, das in der Option RIDFLD angegeben wird, eine relative Satznummer (RRN - Relative Record Number) enthält. Verwenden Sie diese Option nur mit Dateien, die auf Dateien (Datasets) mit relativen Datensätzen verweisen.

SYSID(systemname)

Gibt den Namen (1 - 4 Zeichen) des Systems an, an das die Anforderung übertragen wird.

Wenn Sie SYSID angeben und sowohl RBA als auch RRN weglassen, müssen Sie auch KEYLENGTH angeben, da dieser Wert in der Ressourcendefinition nicht zu finden ist.

TOKEN(datenbereich)

Gibt eine eindeutige ID für diese DELETE-Anforderung als Vollwort-Binärwert an. Verwenden Sie diese ID, um die Löschanforderung einem Datensatz zuzuordnen, der bei einer vorherigen Anforderung READ UPDATE oder BROWSE mit Option UPDATE zurückgegeben wurde. Der zu verwendende Wert ist der Wert, der in dem TOKEN zurückgegeben wird, das von der früheren Anforderung READ UPDATE oder BROWSE mit Option UPDATE gehalten wird.

TOKEN kann durch Funktionsverlagerung übertragen werden. Wenn jedoch eine Anforderung mit der Angabe TOKEN durch Funktionsverlagerung an ein Mitglied der CICS-Produktfamilie übertragen wird, das diese Option nicht erkennt, schlägt die Anforderung fehl.

Bedingungen

105 CHANGED

RESP2-Werte:

109

Es wurde ein Befehl **DELETE** (ohne RIDFLD) für eine Datei abgesetzt, die als Coupling-Facility-Datentabelle mit dem Konfliktmodell für Aktualisierungen definiert ist, und der Datensatz wurde seit dem Zeitpunkt, zu dem das Anwendungsprogramm sie zur Aktualisierung gelesen hat, geändert. Zur erfolgreichen Ausführung des Befehls **DELETE** können Sie die READ-Operation mit der Option UPDATE wiederholen, um die neueste Version des Datensatzes abzurufen, und anschließend den Befehl **DELETE** erneut versuchen.

Standardaktion: Die Task wird abnormal beendet.

84 DISABLED

RESP2-Werte:

50

Eine Datei ist inaktiviert. Eine Datei kann aus folgenden Gründen inaktiviert sein:

- Sie wurde anfänglich als inaktiviert definiert und wurde seither nicht aktiviert.
- Sie wurde durch einen Befehl **SET FILE** oder **CEMT SET FILE** inaktiviert.

Diese Bedingung kann auftreten, wenn der Befehl **DELETE** auf einen Befehl READ mit der Option UPDATE folgt.

Standardaktion: Die Task wird abnormal beendet.

15 DUPKEY

RESP2-Werte:

140

Der Zugriff auf einen Datensatz erfolgt über einen Alternativindex mit dem Attribut NONUNIQUE-KEY und ein weiterer Alternativindexdatensatz mit demselben Schlüssel folgt.

Standardaktion: Die Task wird abnormal beendet.

12 FILENOTFOUND

RESP2-Werte:

1

Der Dateiname, auf den in der Option FILE verwiesen wird, kann in der Dateiressourcendefinition nicht gefunden werden.

Standardaktion: Die Task wird abnormal beendet.

21 ILLOGIC

RESP2-Werte:

110

Es ist ein VSAM-Fehler aufgetreten, der zu keiner der anderen CICS-Antwortkategorien gehört.

Weitere Informationen sind bei EIBRCODE im EXEC-Schnittstellenblock verfügbar; Details finden Sie unter [EIB-Felder](#).

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

RESP2-Werte:

20

Löschoperationen sind gemäß der Ressourcendefinition nicht zulässig.

21

Ein Befehl DELETE wurde für eine Datei abgesetzt, die auf eine VSAM-ESDS-Datei verweist.

22

Ein Befehl DELETE mit der Option GENERIC wurde für eine Datei abgesetzt, die keine VSAM-KSDS-Datei ist.

25

Die Optionen KEYLENGTH und GENERIC werden angegeben und die in der Option KEYLENGTH angegebene Länge ist größer oder gleich der Länge eines vollständigen Schlüssels.

26

Die Option KEYLENGTH wurde angegeben (jedoch nicht die Option GENERIC) und die angegebene Schlüssellänge ist nicht mit der Länge identisch, die für die Datei definiert ist, auf die sich diese Dateiangabe bezieht.

27

Ein Befehl **DELETE** wurde für eine Datei abgesetzt, die auf eine BDAM-Datei (Dataset) verweist.

31

Ein Befehl **DELETE** wurde ohne Option RIDFLD für eine Datei abgesetzt, für die kein vorheriger Befehl **READ UPDATE** abgesetzt worden war.

42

Die Optionen KEYLENGTH und GENERIC werden angegeben und die in der Option KEYLENGTH angegebene Länge ist kleiner als null.

44

Der Befehl **DELETE** entspricht nicht dem richtigen Format für eine benutzerverwaltete Datentabelle oder Coupling-Facility-Datentabelle, zum Beispiel wenn die Option RBA angegeben wurde.

47

Eine Anweisung DELETE enthält ein Token, dessen Wert mit keinem der Token übereinstimmt, die für eine vorhandene Anforderung READ mit Option UPDATE verwendet werden.

51

Ein Befehl mit Angabe des Schlüsselworts RBA oder XRBA wurde für eine KSDS-Datei abgesetzt, auf die im RLS-Modus zugegriffen wird. RLS unterstützt den Zugriff über relative Byteadressen (RBA) auf KSDS-Dateien nicht.

55

Die Option wurde für eine Nicht-RLS-Datei angegeben.

56

Der Versuch, eine wiederherstellbare Coupling-Facility-Datentabelle zu aktualisieren, ist fehlgeschlagen, weil die aktuelle Arbeitseinheit bereits 1024 wiederherstellbare Coupling-Facility-Datentabellen aktualisiert hat. Innerhalb einer einzigen Arbeitseinheit können nicht mehr als 1024 wiederherstellbare Coupling-Facility-Datentabellen aktualisiert werden.

57

DELETE-Operationen sind nicht zulässig, weil die zugeordnete VSAM-Datei den Status AVAILABILITY von RREPL hat, der solche Operationen mit Ausnahme derer vom Replikationsprogramm nicht zulässt.

59

XRBA wurde angegeben, jedoch ist die Datei (Dataset) keine erweiterte ESDS-Datei.

Standardaktion: Die Task wird abnormal beendet.

17 IOERR

RESP2-Werte:

120

Während der Dateisteuerungsoperation ist ein E/A-Fehler aufgetreten. Ein E/A-Fehler ist ein nicht erwartetes Ereignis, das nicht durch eine CICS-Bedingung abgedeckt ist.

Bei VSAM-Dateien gibt die Bedingung IOERR für gewöhnlich einen Hardwarefehler an. Weitere Informationen sind im EXEC-Schnittstellenblock verfügbar; Details finden Sie unter [EIB-Felder](#).

Bei einer Coupling-Facility-Datentabelle gibt eine Bedingung IOERR an, dass eine falsche Antwort von einem Coupling-Facility-Zugriff zurückgegeben wurde.

Standardaktion: Die Task wird abnormal beendet.

54 ISCINVREQ

RESP2-Werte:

70

Das ferne System gibt einen Fehler an, der keiner bekannten Bedingung entspricht.

Standardaktion: Die Task wird abnormal beendet.

94 LOADING

RESP2-Werte:

104

Eine Löschanforderung wurde für eine benutzerverwaltete Datentabelle abgesetzt, die momentan geladen wird. Eine benutzerverwaltete Datentabelle kann nicht während des Ladens geändert werden.

Die Bedingung LOADING wird auch für eine Coupling-Facility-Datentabelle zurückgegeben, wenn die Löschanforderung für einen Schlüssel gilt, der noch nicht geladen wurde. Eine Coupling-Facility-Datentabelle kann während des Ladens geändert werden, jedoch nur, wenn der angeforderte Schlüssel in dem Bereich der bereits geladenen Datensätze liegt.

Die Antwort LOADING kann außerdem für eine Coupling-Facility-Datentabelle zurückgegeben werden, bei deren Laden ein Fehler aufgetreten ist. Weitere Informationen dazu, was geschieht, wenn das Laden einer Coupling-Facility-Datentabelle fehlschlägt, finden Sie in der Beschreibung des globalen Benutzerexits XD TLC unter [Exits zur Datentabellenverwaltung \(XDTRD, XDTAD und XD TLC\)](#).

Wenn Ihr Anwendungsprogramm persistent oder zu häufig eine Bedingung LOADING feststellt, prüfen Sie, ob diese Bedingung nicht durch widersprüchliche Dateidefinitionen verursacht wird, die auf dieselbe Datei (Dataset) verweisen.

Standardaktion: Die Task wird abnormal beendet.

100 LOCKED

RESP2-Werte:

106

Es wurde versucht, mit der Option RIDFLD einen Datensatz zu löschen, jedoch besteht eine *beibehaltene* Sperre für diesen Schlüssel (siehe „Beibehaltene und aktive Sperren“ auf Seite 128). Wenn die Anforderung das Schlüsselwort GENERIC angibt, werden alle möglichen Datensätze gelöscht, während die gesperrten Datensätze verbleiben. Die Anzahl der gelöschten Datensätze wird in NUMREC zurückgegeben.

Die Bedingung LOCKED kann außerdem für eine Anforderung DELETE für eine wiederherstellbare CFDT auftreten, die das Sperrmodell verwendet, wenn der Datensatz, der gelesen werden soll, durch eine beibehaltene Sperre gesperrt ist. Informationen zur Untersuchung beibehaltener Sperren für Datensätze in einer Coupling-Facility-Datentabelle (CFDT) finden Sie unter Beibehaltene Sperren der Coupling-Facility-Datentabelle.

Standardaktion: Die Task wird mit dem Code AEX8 abnormal beendet.

70 NOTAUTH

RESP2-Werte:

101

Eine Ressourcensicherheitsprüfung ist für FILE(dateiname) fehlgeschlagen.

Standardaktion: Die Task wird abnormal beendet.

13 NOTFND

RESP2-Werte:

80

Ein Versuch, einen Datensatz auf der Basis des angegebenen Sucharguments zu löschen, war nicht erfolgreich.

Für benutzerverwaltete Datentabellen und Coupling-Facility-Datentabellen tritt diese Bedingung auf, wenn ein Versuch, einen Datensatz zu löschen, nicht erfolgreich ist, weil es keinen Eintrag mit dem angegebenen Schlüssel in der Datentabelle gibt. Dies kann bei einem Versuch, mit dem Befehl ohne Option RIDFLD einen Datensatz zu löschen, wenn die Löschung einer Anforderung READ UPDATE für einen Datensatz zugeordnet ist, den diese Transaktion (durch DELETE mit Option RIDFLD) gelöscht hat, nachdem er zur Aktualisierung gelesen wurde.

Dies bedeutet nicht, dass kein solcher Datensatz in der Quelldatei vorhanden ist (sofern die Tabelle aus einer erstellt wurde). Es ist möglich, dass ein solcher Datensatz vorhanden ist, jedoch entweder beim einleitenden Laden durch den Benutzerexit XDTRD zurückgewiesen wurde oder später aus der Datentabelle gelöscht wurde.

Für Coupling-Facility-Datentabellen kann diese Bedingung auch dann auftreten, wenn ein Befehl DELETE (ohne Option RIDFLD) für eine Coupling-Facility-Datentabelle mit dem Konfliktmodell abgesetzt wird und der Datensatz seit dem Lesen zur Aktualisierung gelöscht wurde.

Standardaktion: Die Task wird abnormal beendet.

19 NOTOPEN

RESP2-Werte:

60

Die Bedingung NOTOPEN (RESP2 60) wird aus einem der folgenden Gründe zurückgegeben:

- Die angeforderte Datei ist geschlossen (CLOSED) und nicht aktiviert (UNENABLED). Der Status CLOSED, UNENABLED wird erreicht, wenn eine CLOSE-Anforderung für eine geöffnete aktivierte (OPEN ENABLED) Datei empfangen wurde und die Datei nicht mehr im Gebrauch ist. Sie können

CLOSED, UNENABLED auch zum Anfangsstatus machen, indem Sie STATUS(UNENABLED) und OPENTIME(FIRSTREF) in der FILE-Ressourcendefinition angeben.

- Die angeforderte Datei besitzt den Status OPEN (geöffnet) und wird durch andere Transaktionen verwendet, aber für die Datei wurde eine Anforderung CLOSE empfangen.
- Ein Befehl DELETE wird für eine Datei (Dataset) abgesetzt, die infolge eines Befehls SET DSNAME QUIESCED oder IMMQUIESCED in den Quiescemodus (QUIESCED) versetzt wurde oder gerade wird.
- Die angeforderte Datei hat den Status CLOSED (geschlossen) und ENABLED (aktiviert), weshalb CICS versucht hat, die Datei im Rahmen der Anforderungsausführung zu öffnen. Das Öffnen der Datei ist aus einem nicht näher bezeichneten Grund fehlgeschlagen. Prüfen Sie die Konsole auf Nachrichten, die erläutern, warum das Öffnen der Datei nicht erfolgreich war.

Diese Bedingung tritt nicht auf, wenn die Anforderung für eine geschlossene, nicht aktivierte Datei (CLOSED, DISABLED) abgesetzt wird. In diesem Fall tritt die Bedingung DISABLED auf.

Diese Bedingung kann auch nicht auftreten, wenn ein Datensatz gelöscht wird, der gerade zur Aktualisierung gelesen wurde.

Standardaktion: Die Task wird abnormal beendet.

101 RECORDBUSY

RESP2-Werte:

107

Das Schlüsselwort NOSUSPEND wurde für die Löschung eines Datensatzes angegeben, der durch eine *aktive* VSAM-Sperre gesperrt ist (siehe „Beibehaltene und aktive Sperren“ auf Seite 128).

Wenn die Anforderung das Schlüsselwort GENERIC angibt, werden alle möglichen Datensätze mit Ausnahme der verbleibenden gesperrten Datensätze gelöscht. Die Anzahl der gelöschten Datensätze wird in NUMREC zurückgegeben.

Standardaktion: Die Task wird mit dem Code AEX9 abnormal beendet.

53 SYSIDERR

RESP2-Werte:

130

Die Option SYSID gibt einen Namen an, der weder das lokale System noch ein fernes System (das in CICS durch die Definition einer Verbindung CONNECTION bekannt ist) bezeichnet. Die Bedingung SYSIDERR tritt außerdem auf, wenn die Verbindung zu dem fernen System geschlossen ist.

131

Für eine Coupling-Facility-Datentabelle ist die Verbindung zum Server für Coupling-Facility-Datentabellen fehlgeschlagen. Dieser Fehler kann auftreten, weil der Server selbst ausgefallen ist oder der Server verfügbar ist, jedoch CICS keine Verbindung zu ihm hergestellt hat.

132

Der Befehl **DELETE** wurde für eine Coupling-Facility-Datentabelle abgesetzt, die nicht mehr vorhanden ist, wahrscheinlich aufgrund eines Fehlers der Coupling-Facility. In diesem Fall fällt der Coupling-Facility-Datentabellenserver ebenfalls aus.

Standardaktion: Die Task wird abnormal beendet.

Beibehaltene und aktive Sperren

Die Bedingung RECORDBUSY bezieht sich auf aktive Sperren und die Bedingung LOCKED bezieht sich auf beibehaltene Sperren:

- DELETE-Anforderungen für Datensätze, die beibehaltene Sperren haben, werden immer mit der Antwort LOCKED zurückgewiesen.
- DELETE-Anforderungen für Datensätze, die aktive Sperren haben, warten auf die Freigabe der Sperre, sofern nicht das Schlüsselwort NOSUSPEND angegeben ist. Wenn dies der Fall ist, gibt CICS die Antwort RECORDBUSY zurück.

Beispiele

Das folgende Beispiel zeigt, wie eine Gruppe von Datensätzen aus einer VSAM-Datei gelöscht wird:

```
EXEC CICS DELETE  
      FILE('MASTVSAM')  
      RIDFLD(ACCTNO)  
      KEYLENGTH(1en)  
      GENERIC  
      NUMREC(NUMDEL)
```

DELETE ACTIVITY

Löscht eine untergeordnete BTS-Aktivität.

DELETE

➤ DELETE — ACTIVITY(*datenwert*) ➤

Bedingungen: ACTIVITYBUSY, ACTIVITYERR, INVREQ, IOERR, LOCKED

Beschreibung

Der Befehl DELETE ACTIVITY entfernt eine untergeordnete Aktivität aus der Repository-Datei, in der sie definiert ist. Das Beendigungsereignis der untergeordneten Aktivität wird aus dem Ereignispool der übergeordneten Aktivität entfernt. Alle weiteren untergeordneten Aktivitäten der untergeordneten Aktivität werden ebenfalls gelöscht.

Die zu löschende Aktivität muss eine untergeordnete Aktivität der Aktivität sein, die den Befehl DELETE absetzt. Um zur Löschung in Frage zu kommen, muss sich die untergeordnete Aktivität einen der folgenden Verarbeitungsstatuswerte (Modi) haben:

- COMPLETE: Normal oder abnormal beendet oder zuvor abgebrochen.
- INITIAL: Noch nicht ausgeführt oder durch einen Befehl RESET ACTIVITY zurückgesetzt.

Eine Beschreibung aller möglichen Verarbeitungsstatuswerte finden Sie unter [Verarbeitungsmodi](#).

Anmerkung: Eine untergeordnete Aktivität, die nicht explizit durch einen Befehl DELETE ACTIVITY gelöscht wird, wird automatisch durch CICS gelöscht, wenn die übergeordnete Aktivität abgeschlossen wird.

Optionen

ACTIVITY(datenwert)

Gibt den Namen (1-16 Zeichen) der untergeordneten Aktivität an, die gelöscht werden soll.

Bedingungen

107 ACTIVITYBUSY

RESP2-Werte:

19

Die Anforderung hat das Zeitlimit überschritten. Es kann sein, dass die Beendigung einer anderen Task, die diesen Aktivitätsdatensatz verwendet, verhindert wurde.

109 ACTIVITYERR

RESP2-Werte:

8

Die in der Option ACTIVITY angegebene Aktivität konnte nicht gefunden werden.

14

Die in der Option ACTIVITY angegebene untergeordnete Aktivität ist nicht im Modus COMPLETE oder INITIAL und kann dementsprechend nicht gelöscht werden.

16 INVREQ

RESP2-Werte:

4

Der Befehl DELETE ACTIVITY wurde außerhalb des Gültigkeitsbereichs einer momentan aktiven Aktivität abgesetzt.

17 IOERR

RESP2-Werte:

29

Die Repository-Datei ist nicht verfügbar.

30

Es ist ein Ein-/Ausgabefehler für die Repository-Datei aufgetreten.

100 LOCKED

Die Anforderung kann nicht ausgeführt werden, weil eine Sperre für den relevanten Datensatz in der Repository-Datei besteht.

DELETE CHANNEL

Löscht einen benannten Kanal und alle Container, die sich in ihm befinden.

DELETE CHANNEL

► DELETE CHANNEL(*datenwert*) ◄

Bedingungen: CHANNELERR

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl **DELETE CHANNEL** löscht den angegebenen Kanal und alle Container, die sich in dem Kanal befinden. Wenn Sie einen Kanal und seine Container löschen, geschieht Folgendes:

- Alle Daten, die sich in den Containern befinden, werden verworfen.
- Sämtlicher Speicher, der zu dem Kanal und seinen Containern gehört, wird freigegeben.

Das Anwendungsprogramm, das den Befehl **DELETE CHANNEL** absetzt, muss das Programm sein, das Eigner des Kanals ist. Das Programm, das Eigner des Kanals ist, ist das Programm, das den Kanal in einem der folgenden Befehle benannt hat:

- **LINK PROGRAM CHANNEL**
- **MOVE CONTAINER CHANNEL TOCHANNEL**
- **PUT CONTAINER CHANNEL**
- **PUT64 CONTAINER**
- **RETURN TRANSID CHANNEL**
- **START TRANSID CHANNEL**
- **XCTL PROGRAM CHANNEL**
- **WEB RECEIVE TOCHANNEL**
- **WEB CONVERSE TOCHANNEL**

Ein Anwendungsprogramm kann die folgenden Kanäle nicht löschen:

- Den aktuellen Kanal für das Anwendungsprogramm, das heißt, den Kanal, mit dem das Programm aufgerufen wurde.

- Jeden Kanal, der nicht von dem Anwendungsprogramm erstellt wurde.
- Jeden schreibgeschützten Kanal.
- Den Transaktionskanal DFHTRANSACTION.

Optionen

CHANNEL(datenwert)

Gibt den 1 bis 16 Zeichen langen Namen des Kanals an, der gelöscht werden soll. Alle Container, deren Eigner der Kanal ist, werden gelöscht und der Kanal selbst wird gelöscht.

Bedingungen

122 CHANNELERR

RESP2-Werte:

2

Der in der Option CHANNEL angegebene Kanal konnte nicht gefunden werden.

3

Der in der Option CHANNEL angegebene Kanal ist schreibgeschützt.

4

Der in der Option CHANNEL angegebene Kanal ist der aktuelle Kanal.

5

Der in der Option CHANNEL angegebene Kanal ist der Transaktionskanal.

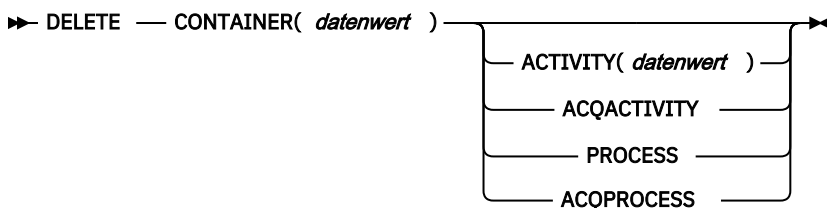
6

Der in der Option CHANNEL angegebene Kanal gehört nicht dem aufrufenden Programm.

DELETE CONTAINER (BTS)

Löscht einen benannten BTS-Datencontainer.

DELETE CONTAINER (BTS)



Bedingungen: ACTIVITYERR, CONTAINERERR, INVREQ, IOERR, LOCKED, PROCESSBUSY

Beschreibung

Der Befehl DELETE CONTAINER (BTS) löscht einen BTS-Datencontainer und verwirft alle Daten, die er enthält.

Der Container wird durch den Namen und durch den Prozess oder die Aktivität identifiziert, für den bzw. die er ein Container ist, d. h. der Prozess oder die Aktivität ist "Eigner" des Containers. Die Aktivität, die Eigner des Containers ist, kann wie folgt angegeben werden:

- Explizit durch Angabe einer der Optionen für Prozess oder Aktivität.
- Implizit durch Weglassen der Optionen für Prozess oder Aktivität. Wenn diese Optionen nicht angegeben werden, wird die aktuelle Aktivität impliziert.

Anmerkung: Prozesscontainer können nur durch die Stammaktivität oder durch ein Programm, das den Prozess angefordert hat, gelöscht werden.

Optionen

ACQACTIVITY

Gibt eine der folgenden Informationen an:

- Wenn das Programm, das den Befehl absetzt, einen Prozess angefordert hat, gibt diese Option an, dass der Eigner des Containers die Stammaktivität dieses Prozesses ist.
- Andernfalls gibt diese Option an, dass der Container der Aktivität gehört, die von dem Programm durch einen Befehl ACQUIRE ACTIVITYID angefordert wurde.

ACQPROCESS

Gibt an, dass der Container dem Prozess gehört, den das Programm, das den Befehl absetzt, in der aktuellen Arbeitseinheit angefordert hat.

ACTIVITY(datenwert)

Gibt den Namen (1-16 Zeichen) der Aktivität an, die Eigner des Containers ist. Dies muss eine untergeordnete Aktivität der aktuellen Aktivität sein.

CONTAINER(datenwert)

Gibt den Namen (1-16 Zeichen) des Containers an, der gelöscht werden soll.

PROCESS

Gibt an, dass der zu löschende Container dem aktuellen Prozess gehört, das heißt, dem Prozess, für den das Programm ausgeführt wird, das den Befehl absetzt.

Bedingungen

109 ACTIVITYERR

RESP2-Werte:

8

Die in der Option ACTIVITY angegebene Aktivität konnte nicht gefunden werden.

110 CONTAINERERR

RESP2-Werte:

10

Der in der Option CONTAINER angegebene Container konnte nicht gefunden werden.

26

Der in der Option CONTAINER angegebene Prozesscontainer ist schreibgeschützt. (Prozesscontainer können nur durch die Stammaktivität oder durch ein Programm, das den Prozess angefordert hat, gelöscht werden.)

16 INVREQ

RESP2-Werte:

4

Der Befehl wurde außerhalb des Gültigkeitsbereichs einer momentan aktiven Aktivität abgesetzt.

15

Die Option ACQPROCESS wurde verwendet, jedoch hat die Arbeitseinheit, die die Anforderung abgesetzt hat, keinen Prozess angefordert.

24

Die Option ACQACTIVITY wurde verwendet, jedoch hat die Arbeitseinheit, die die Anforderung abgesetzt hat, keine Aktivität angefordert.

25

Die Option PROCESS wurde verwendet, jedoch wurde der Befehl außerhalb des Gültigkeitsbereichs eines momentan aktiven Prozesses abgesetzt.

17 IOERR

RESP2-Werte:

30

Es ist ein Ein-/Ausgabefehler für die Repository-Datei aufgetreten.

10

Der in der Option CONTAINER angegebene Container konnte nicht gefunden werden.

16 INVREQ

RESP2-Werte:

4

Der Befehl wurde außerhalb des Gültigkeitsbereichs des momentan aktiven Kanals abgesetzt.

30

Sie können einen von CICS definierten schreibgeschützten Container nicht löschen.

DELETE COUNTER und DELETE DOUNTER

Löscht den benannten Zähler aus dem angegebenen Pool. COUNTER wird für Vollwortzähler mit Vorzeichen verwendet, DOUNTER für Doppelwortzähler ohne Vorzeichen.

DELETE COUNTER

➔ DELETE COUNTER(*name*) ————
 POOL(*name*) NOSUSPEND

Bedingungen: BUSY, INVREQ

Dieser Befehl ist threadsicher.

DELETE DOUNTER

➔ DELETE DOUNTER(*name*) ————
 POOL(*name*) NOSUSPEND

Bedingungen: BUSY, INVREQ

Dieser Befehl ist threadsicher.

Beschreibung

Diese Befehle löschen den angegebenen Zähler aus dem angegebenen Pool.

Optionen

COUNTER(*name*)

Gibt den Namen des zu löschenden Vollwortzählers an. Der Name kann bis zu 16 alphanumerische Zeichen lang sein. Falls es sich bei *name* um eine Variable handelt, die einen weniger als 16 Zeichen langen Namen enthält, muss der Name mit nachgestellten Leerzeichen aufgefüllt werden.

DOUNTER(*name*)

Gibt den Namen des zu löschenden Doppelwortzählers an. Der Name kann bis zu 16 alphanumerische Zeichen lang sein. Falls es sich bei *name* um eine Variable handelt, die einen weniger als 16 Zeichen langen Namen enthält, muss der Name mit nachgestellten Leerzeichen aufgefüllt werden.

NOSUSPEND

Gibt eine sofortige Rückkehr zum Anwendungsprogramm während einer Neuerstellung der Coupling-Facility-Struktur an. Wenn Sie den Parameter NOSUSPEND nicht angeben, wird die Anforderung so lange wiederholt, bis sie erfolgreich ist. Wenn Sie NOSUSPEND angeben, schlägt die Anforderung sofort fehl und CICS gibt die Bedingung BUSY, RESP2=500 zurück.

POOL(*poolname*)

Gibt eine acht Zeichen lange Zeichenfolge an, die als Poolauswahlparameter verwendet werden soll, um den Pool auszuwählen, in dem sich der benannte Zähler befindet. Die Zeichenfolge kann ein logischer Poolname oder der tatsächliche Poolname sein.

Gültige Zeichen für die Poolselektorseichenfolge sind A bis Z, 0 bis 9, \$ @ # und _ (Unterstrichzeichen). Falls es sich bei *name* um eine Variable handelt, die einen weniger als 8 Zeichen langen Namen enthält, muss der Name mit nachgestellten Leerzeichen aufgefüllt werden.

Dieser Parameter ist optional. Wenn Sie den Namen des Pools nicht angeben, wird ein Poolauswahlwert aus 8 Leerzeichen angenommen.

Wenn kein entsprechender Eintrag in der Optionstabelle DFHNCOPT vorhanden ist, verwendet CICS den Namen, der im Systeminitialisierungsparameter NCPLDFT angegeben ist, der den Standardpool für benannte Zähler angibt.

Informationen zum Generieren einer Optionstabelle für benannte Zähler mithilfe des Makros DFHNCO finden Sie unter Server für benannte Zähler.

Bedingungen

128 BUSY

RESP2-Werte:

500

Die Option NOSUSPEND wurde im Befehl angegeben und die Coupling-Facility-Struktur ist momentan während einer Neuerstellung nicht verfügbar.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

RESP2-Werte:

201

Der benannte Zähler wurde nicht gefunden.

301

Der Server hat einen Fehlercode gemeldet, der für die Schnittstelle des benannten Zählers unverständlich ist. Dies ist in der Regel nur dann möglich, wenn das Schnittstellenlademodul DFHNCIF eine geringere Wartungsstufe oder ein niedrigeres Release-Level als der Server selbst aufweist.

303

Bei einem Makro, das für den Zugriff auf die Coupling-Facility verwendet wird, trat ein unerwarteter Fehler wie beispielsweise eine Strukturstörung oder eine Verbindungsunterbrechung auf. Weitere Informationen enthält die Nachricht DFHNC0441 im Jobprotokoll der Anwendung.

304

Der im Programm angegebene Parameter für die Poolauswahl kann anhand der aktuellen Optionstabelle nicht in einen gültigen Servernamen aufgelöst werden.

305

Die Schnittstelle kann keine Verbindung zum Server für den ausgewählten Pool mit benannten Zählern herstellen. Weitere Informationen enthält eine AXM-Servicenachricht (AXMSCnnnn) im Jobprotokoll der Anwendung.

306

Während der Server eine Anforderung verarbeitete, trat eine abnormale Beendigung auf. Weitere Informationen enthält eine Nachricht im Jobprotokoll der Anwendung und im Jobprotokoll des Servers.

308

Das Optionstabellenmodul DFHNCOPT, das zum Auflösen eines Poolnamens benötigt wird, konnte nicht geladen werden.

309

Während der Verarbeitung der Optionstabelle hat die Schnittstelle für benannte Zähler ein unbekanntes Eintragsformat festgestellt. Entweder wurde die Optionstabelle nicht ordnungsgemäß generiert oder das Schnittstellenlademodul DFHNCIF hat nicht dasselbe Release-Level wie die Optionstabelle.

310

Ein Optionstabelleneintrag, der mit dem angegebenen Poolnamen übereinstimmt, hat ein Benutzerexitprogramm angegeben, aber das Benutzerexitprogramm ist nicht durch eine Linkbearbeitung mit der Optionstabelle verbunden und kann nicht geladen werden.

311

Eine Antwort vom Server des benannten Zählers an das Schnittstellenmodul DFHNCIF der Client-region gibt an, dass gegenwärtig ein systemverwalteter erneuter Build ausgeführt wird, aber die Bedingung wird von der EXEC CICS-Schnittstelle nicht erkannt. Dies bedeutet, dass die CICS-Region den Stand von CICS TS 2.1 oder früher aufweist.

403

Der Parameter POOL enthält ungültige Zeichen oder eingebettete Leerzeichen.

Standardaktion: Die Task wird abnormal beendet.

DELETE EVENT

Löscht ein BTS-Ereignis.

DELETE EVENT

➡ DELETE — EVENT(*datenwert*) ➡

Bedingungen: EVENTERR, INVREQ

Beschreibung

Der Befehl DELETE EVENT löscht ein BTS-Ereignis, das nicht mehr benötigt wird. Das Ereignis wird aus dem Ereignispool der aktuellen Aktivität entfernt. Ein Ereignis kann unabhängig davon gelöscht werden, ob es ausgelöst wurde oder nicht.

Der Befehl DELETE EVENT kann nur zum Löschen der folgenden Typen von Ereignissen verwendet werden:

- Eingabeereignis
- Zusammengesetztes Ereignis

Der Befehl DELETE EVENT kann *nicht* zum Löschen der folgenden Typen von Ereignissen verwendet werden:

- Aktivitätsbeendigungsereignisse. Diese werden implizit gelöscht, wenn eine Antwort aus der abgeschlossenen Aktivität durch einen Befehl CHECK ACTIVITY bestätigt wird, der durch die übergeordnete Aktivität der Aktivität abgesetzt wird, oder wenn ein Befehl DELETE ACTIVITY abgesetzt wird.
- Zeitgeberereignisse. Diese werden implizit gelöscht, wenn der Ablauf des zugeordneten Zeitgebers durch einen Befehl CHECK TIMER bestätigt wird oder wenn ein Befehl DELETE TIMER abgesetzt wird.
- Systemereignisse.

Anmerkung:

1. Wenn das zu löschende Ereignis in das Prädikat eines zusammengesetzten Ereignisses eingeschlossen ist, wird es aus dem booleschen Ausdruck des Prädikats entfernt. Der Ausführungsstatus des zusammengesetzten Ereignisses (FIRED oder NOTFIRED) wird erneut ausgewertet.
2. Das Löschen eines zusammengesetzten Ereignisses hat keine Auswirkung auf die zugehörigen Unterereignisse.

Optionen

EVENT(datenwert)

Gibt den Namen (1-16 Zeichen) des Ereignisses an, das gelöscht werden soll.

Bedingungen

111 EVENTERR

RESP2-Werte:

4

Das in der Option EVENT angegebene Ereignis wird von BTS nicht erkannt.

16 INVREQ

RESP2-Werte:

1

Der Befehl wurde außerhalb des Gültigkeitsbereichs einer Aktivität abgesetzt.

2

Das in der Option EVENT angegebene Ereignis kann nicht gelöscht werden, weil es ein Systemereignis, ein Zeitgeberereignis oder ein Aktivitätsbeendigungsereignis ist.

DELETE TIMER

Löscht einen BTS-Zeitgeber.

DELETE TIMER

➤ DELETE — TIMER(*datenwert*) ➤

Bedingungen: INVREQ, TIMERERR

Beschreibung

Der Befehl DELETE TIMER löscht einen BTS-Zeitgeber. Wenn dem Zeitgeber ein Ereignis zugeordnet ist, wird auch das Ereignis gelöscht und aus dem Ereignispool der aktuellen Aktivität entfernt. (Dem Zeitgeber ist kein Ereignis zugeordnet, wenn der Zeitgeber abgelaufen ist und ein Befehl CHECK TIMER abgesetzt wurde.)

Die einzigen Zeitgeber, die ein Programm löschen kann, sind diejenigen, deren Eigner die aktuelle Aktivität ist. Ein Zeitgeber kann unabhängig davon gelöscht werden, ob er abgelaufen ist oder nicht.

Optionen

TIMER(*datenwert*)

Gibt den Namen (1 - 16 Zeichen) des Zeitgebers an, der gelöscht werden soll.

Bedingungen

16 INVREQ

RESP2-Werte:

1

Der Befehl wurde außerhalb des Gültigkeitsbereichs einer momentan aktiven Aktivität abgesetzt.

115 TIMERERR

RESP2-Werte:

13

Der Zeitgeber, der in der Option TIMER angegeben wurde, ist nicht vorhanden.

DELETEQ TD

Löscht alle transienten Daten.

DELETEQ TD

➤ DELETEQ TD — QUEUE(*name*) — SYSID(*systemname*) ➤

Bedingungen: DISABLED, INVREQ, ISCINVREQ, LOCKED, NOTAUTH, QIDERR, SYSIDERR

Dieser Befehl ist threadsicher, wenn er mit einer Warteschlange in einer lokalen CICS-Region verwendet wird oder wenn er durch Funktionsverlagerung über eine IPIC-Verbindung an eine ferne CICS-Region übertragen wird. Er ist nicht threadsicher, wenn er durch Funktionsverlagerung über einen anderen Typ von Verbindung an eine ferne CICS-Region übertragen wird.

Beschreibung

Der Befehl DELETEQ TD löscht alle transienten Daten, die zu einer bestimmten partitionsinternen Zielwarteschlange gehören. Der gesamte Speicher, der der Zielwarteschlange zugeordnet ist, wird freigegeben. Beachten Sie, dass Sie diesen Befehl nicht verwenden können, um eine *partitionsübergreifende* Warteschlange mit transienten Daten zu löschen. Ein entsprechender Versuch hat eine Bedingung INVREQ zu Folge.

Optionen

QUEUE(*name*)

Gibt den symbolischen Namen (1 - 4 alphanumerische Zeichen) der Warteschlange an, die gelöscht werden soll. Die benannte Warteschlange muss in CICS definiert sein.

Wenn SYSID angegeben wird, wird angenommen, dass sich die Warteschlange auf einem fernen System befindet, und zwar unabhängig davon, wie sie definiert ist. Ansonsten wird die Ressourcendefinition verwendet, um zu ermitteln, ob sich die Warteschlange auf einem lokalen oder fernen System befindet.

SYSID(*systemname*)

(Nur ferne Systeme) Gibt den Namen (1 - 4 Zeichen) des Systems an, an das die Anforderung übertragen wird.

Bedingungen

84 DISABLED

Tritt auf, wenn die Warteschlange inaktiviert wurde.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

Tritt auf, wenn DELETEQ eine partitionsübergreifende Warteschlange angibt.

Standardaktion: Die Task wird abnormal beendet.

54 ISCINVREQ

Tritt auf, wenn das ferne System einen Fehler angegeben hat, der keiner bekannten Bedingung entspricht.

Standardaktion: Die Task wird abnormal beendet.

100 LOCKED

Tritt auf, wenn die Anforderung nicht ausgeführt werden kann, weil die Verwendung der Warteschlange aufgrund einer fehlgeschlagenen unbestätigten Arbeitseinheit eingeschränkt wurde. Dies kann bei einer beliebigen Anforderung für eine logisch wiederherstellbare Warteschlange geschehen, die mit WAIT(YES) und WAITACTION(REJECT) in der TDQUEUE-Ressourcendefinitionen definiert ist.

Geben Sie WAIT(YES) und WAITACTION(QUEUE) in der TDQUEUE-Ressourcendefinition an, wenn die Transaktion warten soll.

Standardaktion: Die Task wird abnormal beendet.

70 NOTAUTH

Tritt auf, wenn eine Ressourcensicherheitsprüfung für QUEUE(*name*) fehlgeschlagen ist.

Standardaktion: Die Task wird abnormal beendet.

44 QIDERR

Tritt auf, wenn das symbolische Ziel, das mit DELETEQ TD verwendet werden soll, nicht gefunden werden kann.

Standardaktion: Die Task wird abnormal beendet.

53 SYSIDERR

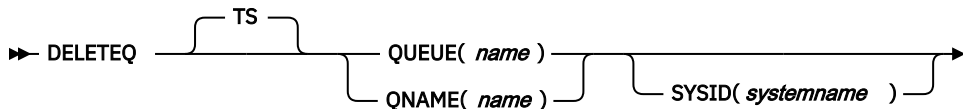
Tritt auf, wenn die Option SYSID einen Namen angibt, der weder das lokale System noch das ferne System (das in CICS durch die Definition einer Verbindung CONNECTION oder IPCONN bekannt ist) bezeichnet. Die Bedingung SYSIDERR tritt außerdem auf, wenn die Verbindung zu dem fernen System geschlossen ist.

Standardaktion: Die Task wird abnormal beendet.

DELETEQ TS

Löscht eine Warteschlange für temporären Speicher.

DELETEQ TS



Bedingungen: INVREQ, ISCINVREQ, LOCKED, NOTAUTH, QIDERR, SYSIDERR

Dieser Befehl ist threadsicher, wenn er zusammen mit einer Warteschlange im Hauptspeicher oder im Hilfsspeicher entweder in einer lokalen CICS-Region verwendet wird oder wenn er durch Funktionsverlagerung über eine IPIC-Verbindung an eine ferne CICS-Region übertragen wird. Er ist außerdem threadsicher, wenn er mit einer Warteschlange in einem gemeinsam genutzten temporären Speicherpool in einer z/OS-Coupling-Facility verwendet wird, die durch einen Server für gemeinsame Datennutzung mit temporären Speicher (TS-Server) verwendet wird. Der Befehl ist nicht threadsicher, wenn er durch Funktionsverlagerung über einen anderen Verbindungstyp als IPIC in eine ferne CICS-Region übertragen wird.

Hinweis für dynamisches Transaktionsrouting: Die Verwendung dieses Befehls kann transaktionsübergreifende Affinitäten erzeugen, die die Nutzung des dynamischen Transaktionsroutings beeinträchtigen. Weitere Informationen zu Transaktionsaffinitäten finden Sie unter [Affinität](#).

Beschreibung

Der Befehl **DELETEQ TS** löscht alle temporären Daten, die zu einer Warteschlange für temporären Speicher gehören. Der gesamte Speicher, der Warteschlange zugeordnet ist, wird freigegeben.

Sie sollten temporäre Daten so bald wie möglich löschen, um die Verwendung von übermäßig viel Speicher zu vermeiden.

Wenn eine wiederherstellbare Warteschlange für temporären Speicher gelöscht wird, müssen Sie einen Synchronisationspunkt absetzen, bevor Sie einen nachfolgenden Befehl **WRITEQ TS** für dieselbe Warteschlange absetzen.

Optionen

QNAME(name)

Als Alternative zur Option QUEUE gibt QNAME den symbolischen Namen (1 - 16 Zeichen) der Warteschlange an, die gelöscht werden soll. Der Name darf nicht nur aus binären Nullen bestehen und muss im CICS-System eindeutig sein. Wenn der Name weniger als 16 Zeichen lang ist, müssen Sie trotzdem ein 16 Zeichen langes Feld verwenden, das mit Leerzeichen aufzufüllen ist, falls dies erforderlich ist.

QUEUE(name)

Gibt den symbolischen Namen (1 - 8 Zeichen) der Warteschlange an, die gelöscht werden soll. Der Name kann nicht nur aus binären Nullen bestehen und muss im CICS-System eindeutig sein. Wenn der Name weniger als 8 Zeichen lang ist, müssen Sie trotzdem ein 8 Zeichen langes Feld verwenden, das mit Leerzeichen aufzufüllen ist, falls dies erforderlich ist.

SYSID(systemname)

(Nur ferne und gemeinsam genutzte Warteschlangen) Gibt den Systemnamen (1-4 Zeichen) an, der das ferne System oder einen Pool von gemeinsam genutzten Warteschlangen bezeichnet, an das bzw.

den die Anforderung übertragen wird. Beachten Sie, dass TSMODEL-Ressourcendefinitionen die Angabe einer System-ID (SYSID) für eine Warteschlange, die sich in einem Pool für gemeinsame Datennutzung des temporären Speichers befindet, nicht unterstützen. Verwenden Sie stattdessen die Option QUEUE oder die Option QNAME. Die Verwendung einer expliziten Angabe von SYSID für einen gemeinsam genutzten Warteschlangenpool setzt die Unterstützung einer Tabelle für temporären Speicher (TST) voraus.

Bedingungen

16 INVREQ

Tritt in einer der folgenden Situationen auf:

- Die Warteschlange wurde durch internen CICS-Code erstellt.
- Der angegebene Warteschlangenname besteht nur aus binären Nullen.

Standardaktion: Die Task wird abnormal beendet.

54 ISCINVREQ

Tritt auf, wenn das ferne System einen Fehler angegeben hat, der keiner bekannten Bedingung entspricht.

Standardaktion: Die Task wird abnormal beendet.

100 LOCKED

Tritt auf, wenn die Anforderung nicht ausgeführt werden kann, weil die Verwendung der Warteschlange aufgrund einer fehlgeschlagenen unbestätigten Arbeitseinheit eingeschränkt wurde.

Standardaktion: Die Task wird abnormal beendet.

70 NOTAUTH

Tritt auf, wenn eine Ressourcensicherheitsprüfung für *QUEUE(name)* fehlgeschlagen ist.

Standardaktion: Die Task wird abnormal beendet.

44 QIDERR

Tritt auf, wenn die angegebene Warteschlange weder im Hauptspeicher noch im Hilfsspeicher gefunden werden kann.

Standardaktion: Die Task wird abnormal beendet.

53 SYSIDERR

Tritt in einer der folgenden Situationen auf:

- Die Option SYSID gibt einen Namen an, der weder das lokale System noch ein fernes System (das in CICS durch die Definition einer Verbindung CONNECTION oder IPCONN bekannt ist) bezeichnet.
- Wenn IPIC-Konnektivität verwendet wird, sind das lokale System, der ferne System oder beide keine Regionen mit CICS TS 4.2 oder höher.
- Die Verbindung zum fernen System wurde geschlossen.
- Die CICS-Region, in der der Befehl für temporären Speicher ausgeführt wird, kann keine Verbindung zu dem TS-Server herstellen, der den TS-Pool verwaltet, der die angegebene Warteschlange für temporären Speicher unterstützt. Diese Situation kann zum Beispiel auftreten, wenn die CICS-Region für den Zugriff auf den Server für temporären Speicher nicht berechtigt ist.

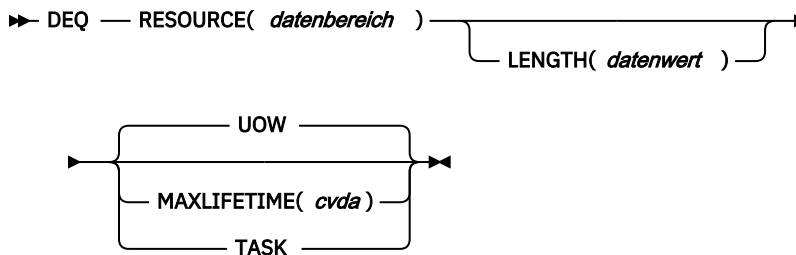
Diese Bedingung kann darüber hinaus auftreten, wenn der Server für temporären Speicher nicht gestartet ist oder weil der Server einen Fehler festgestellt hat (oder gestoppt wurde), während CICS weiterhin ausgeführt wird.

Standardaktion: Die Task wird abnormal beendet.

DEQ

Terminiert die Verwendung einer Ressource durch eine Task (Entfernen aus der Warteschlange).

DEQ



Bedingungen: INVREQ, LENGERR

Dieser Befehl ist threadsicher, wenn er als lokal definiert wird. Er ist nicht threadsicher, wenn er als global definiert wird.

Hinweis für dynamisches Transaktionsrouting: Die Verwendung dieses Befehls kann möglicherweise transaktionsübergreifende Affinitäten erstellen, die sich negativ auf die Verwendung des dynamischen Transaktionsroutings auswirken. Dies gilt nicht, wenn der in RESOURCE angegebene Name mit dem Namen übereinstimmt, der in einer installierten ENQMODEL-Ressourcendefinition angegeben ist, die für den gesamten Sysplex-Bereich gilt. Weitere Informationen zu Transaktionsaffinitäten finden Sie unter [Affinität](#).

Beschreibung

Der Befehl DEQ bewirkt, dass eine Ressource, für die sich die Task zurzeit in die Warteschlange eingereiht hat, zur Verwendung durch andere Tasks freigegeben wird.

Wenn sich eine Task für eine Ressource in die Warteschlange einreicht, sich jedoch nicht aus der Warteschlange entfernt, gibt CICS die Ressource automatisch bei der Synchronisationspunktverarbeitung oder bei Beendigung der Task frei. Eine Ressource im Kontext dieses Befehls ist jede 1 bis 255 Byte lange Zeichenfolge, die durch unternehmensinterne Standards definiert wurde, um gegen widersprüchliche Aktionen zwischen Tasks zu schützen oder um nur einen Thread innerhalb eines Programms zu verwenden.

Beim Absetzen des Befehls DEQ muss die Ressource, für die die Task aus der Warteschlange entfernt werden soll, durch die Methode angegeben werden, die beim Einreihen der Task in die Warteschlange für die Ressource verwendet wurde. Wenn für die Ressource keine Einreihung (ENQ) abgesetzt wurde, wird das Entfernen aus der Warteschlange ignoriert.

Wenn mehr als ein Befehl ENQ für eine Ressource durch eine Task abgesetzt wurde, bleibt die Ressource so lange Eigentum der Task, bis die Task eine entsprechende Anzahl von Befehlen DEQ abgesetzt hat.

Wenn ein Befehl EXEC CICS DEQ (oder ENQ) für eine Ressource abgesetzt wird, deren Name dem einer installierten ENQMODEL-Ressourcendefinition entspricht, prüft CICS den Wert des Attributs ENQSCOPE, um zu ermitteln, ob der Bereich lokal oder der gesamte Sysplex ist. Wenn das Attribut ENQSCOPE nicht leer ist, verarbeitet CICS die Befehle ENQ oder DEQ für den gesamten Sysplex und übergibt den Warteschlangennamen und den Ressourcennamen an z/OS Global Resource Serialization, um die Einreihung zu verwalten. Falls das Attribut ENQSCOPE leer gelassen wurde (Standardwert), verarbeitet CICS den Befehl DEQ als lokal für die absetzende CICS-Region. Wenn keine ENQMODEL-Definition dem Ressourcennamen entspricht, ist der Bereich des Befehls DEQ lokal. Weitere Informationen zu ENQMODEL-Ressourcen finden Sie unter [ENQMODEL-Ressourcen](#).

Optionen

LENGTH(datenwert)

Gibt an, dass die Ressource, für die die Task aus der Warteschlange entfernt werden soll, eine Länge hat, die durch den Datenwert angegeben wird. Der Datenwert ist ein Halbwort-Binärwert aus dem Bereich von 1 bis 255. Wenn der von Ihnen angegebene Wert außerhalb dieses Bereichs liegt, tritt eine Bedingung LENGERR auf. Wenn die Option LENGTH in einem Befehl ENQ angegeben wird, muss sie auch im Befehl DEQ für diese Ressource angegeben werden und die Werte dieser Optionen müssen übereinstimmen.

MAXLIFETIME(*cvda*)

Gibt die Dauer der Einreihung (ENQ) an, die freigegeben wird. Die CVDA-Werte lauten wie folgt:

UOW

Die Einreihung (Enqueue) wurde mit der Dauer einer Arbeitseinheit (UOW) angefordert. Dies ist der Standardwert.

Anmerkung: Aus Gründen der Kompatibilität mit früheren Releases von CICS wird auch der CVDA-Wert LUW unterstützt.

TASK

Die Einreihung (Enqueue) wurde mit der Dauer einer Task angefordert.

RESOURCE(*datenbereich*)

Gibt entweder einen Bereich, dessen Adresse die Ressource darstellt, für die die Task aus der Warteschlange entfernt werden soll, oder eine Variable an, die die Ressource (z. B. einen Mitarbeiternamen) enthält. In letzteren Fall müssen Sie die Option LENGTH verwenden.

Bedingungen**16 INVREQ**

RESP2-Werte:

2

Die Option MAXLIFETIME ist mit einem falschen CVDA-Wert festgelegt.

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

RESP2-Werte:

1

Der Wert, den Sie für die Option LENGTH angegeben haben, liegt außerhalb des Bereichs von 1 bis 255.

Standardaktion: Die Task wird abnormal beendet.

Beispiele

Die folgenden Beispiele zeigen, wie eine Task aus der Warteschlange für eine Ressource entfernt wird:

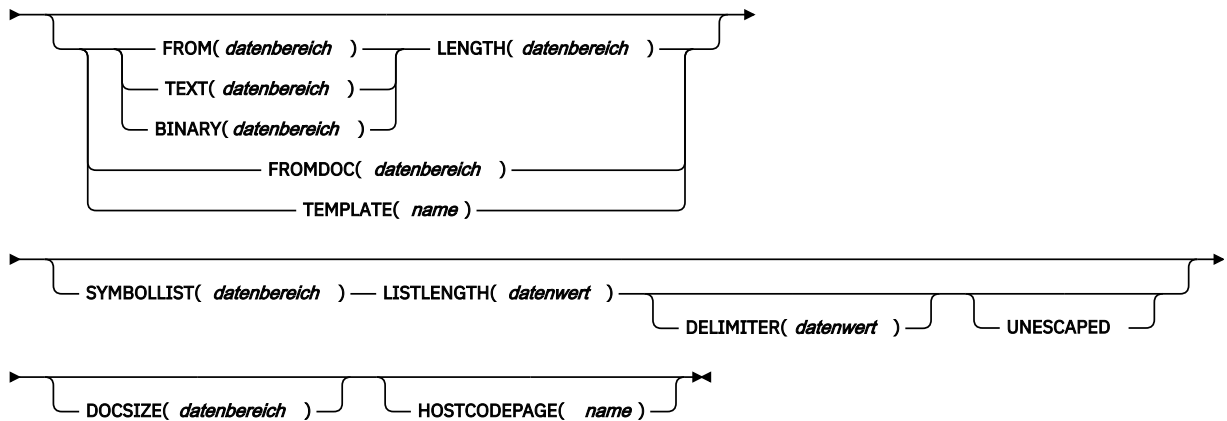
```
EXEC CICS DEQ  
      RESOURCE (RESNAME)  
  
EXEC CICS DEQ  
      RESOURCE (SOCSECNO)  
      LENGTH (9)
```

DOCUMENT CREATE

Erstellt ein Dokument.

DOCUMENT CREATE

➔ DOCUMENT — CREATE — DOCTOKEN(*datenbereich*) →



Bedingungen: INVREQ, LENGERR, NOTAUTH, NOTFND, SYMBOLERR, TEMPLATERR

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl DOCUMENT CREATE signalisiert den Start des Dokumenterstellungsprozesses. Das Dokument, das erstellt werden soll, kann ein leeres Dokument sein oder es kann auf einem vorhandenen Dokument, auf einer Vorlage (TEMPLATE) oder auf Daten, die in einem Anwendungspuffer enthalten sind, basieren.

Optionen

BINARY(datenbereich)

Gibt einen Puffer mit Daten an, die als Inhalt des neu zu erstellenden Dokuments verwendet werden sollen. Die Daten werden unverändert in den Dokumentinhalt kopiert. Es erfolgt kein Versuch, die Daten für die Symbolsubstitution zu parsen. Der Zweck der Option BINARY besteht darin, zuzulassen, dass die Anwendung Blöcke von Daten einfügen kann, die keiner Konvertierung in eine Client-Codepage unterzogen werden dürfen, wenn die Daten gesendet werden. Verwenden Sie die Option LENGTH zur Angabe der Länge dieses Puffers.

DELIMITER(datenwert)

Gibt einen optionalen 1-Byte-Wert an, mit dem Symbolname/Wert-Paare im Puffer SYMBOLLIST begrenzt werden. Wenn diese Option nicht angegeben wird, wird standardmäßig ein Et-Zeichen verwendet. Bestimmte Begrenzungszeichen (z. B. das Leerzeichen) sind nicht zulässig und verursachen bei Verwendung sämtlich eine Bedingung INVREQ für den Befehl. Die Regeln sind im Abschnitt [Regeln zur Angabe von Symbolen und Symbollisten](#) aufgeführt.

Bei Verwendung dieser Option muss die Anwendung sicherstellen, dass das Begrenzungszeichen (DELIMITER) nicht als Symbolwert im Puffer SYMBOLLIST vorkommt. Aus diesem Grund sollte die Anwendung keine alphanumerischen oder anderen druckbaren Zeichen als Wert für DELIMITER verwenden.

DOCSIZE(datenbereich)

Gibt einen Datenbereich mit Vollwort-Binärwert an, der mit der aktuellen Größe des Dokuments in Byte aktualisiert wird. Dies ist die maximale Größe des Puffers, der für eine Kopie des Dokuments benötigt wird, wenn ein Befehl RETRIEVE abgesetzt wird.

DOCTOKEN(datenbereich)

Gibt einen Datenbereich an, der das binäre Token des Dokuments aufnehmen soll. Der Bereich muss 16 Byte lang sein und wird auf einen von CICS generierten Namen gesetzt, durch den in späteren Befehlen auf das Dokument verwiesen werden kann.

FROM(datenbereich)

Gibt an, dass die durch die Anwendung angegebenen Daten verwendet werden sollen, um den Inhalt des neuen Dokuments zu erstellen. Der Dateninhalt könnte eine Vorlage oder ein Dokument sein, die bzw. das erstellt und abgerufen wurde. Wenn die Daten eine Vorlage sind, erfolgt eine Symbolsubstitution dort, wo Symbole in der Symboltabelle vorhanden sind. Wenn die Daten ein zuvor abgerufenes Dokument sind, werden die Konvertierungs- und Lesezeichentags, die beim Abruf eingefügt wurden, aus dem Inhalt entfernt und in dem internen Format gespeichert, das für die API-Befehle erforderlich ist. Beachten Sie, dass keine Symbolsubstitution von nicht aufgelösten Symbolen erfolgt, die in einem abgerufenen Dokument enthalten sind. Verwenden Sie die Option LENGTH zur Angabe der Länge dieses Puffers.

FROMDOC(datenbereich)

Gibt das binäre Token (siehe Option **DOCTOKEN**) eines Dokuments an, dessen Inhalt in das neu zu erstellende Dokument kopiert werden soll. Alle Lesezeichen- und Konvertierungstags werden in das neue Dokument kopiert. Die Symboltabelle wird nicht kopiert.

HOSTCODEPAGE(name)

Gibt den Namen der Host-Codepage an, in der die Daten, die hinzugefügt werden, codiert werden. Diese Option gilt nur für die Optionen TEXT, SYMBOL und TEMPLATE. Der Name muss acht Zeichen lang sein. Ist er kürzer als acht Zeichen, muss er auf der rechten Seite mit Leerzeichen aufgefüllt werden.

Das CICS-Standardformat für einen Codepagenamen besteht aus der Codepagenummer (oder allgemeiner der CCSID), die nach Bedarf mithilfe von 3 bis 5 Dezimalziffern geschrieben und anschließend mit nachgestellten Leerzeichen auf 8 Zeichen aufgefüllt wird. Bei der Codepage 37, die aus weniger als 3 Ziffern besteht, lautet das Standardformat '037'. CICS akzeptiert jetzt jede Dezimalzahl aus bis zu acht Ziffern (mit nachfolgenden Leerzeichen aufgefüllt) aus dem Bereich von 1 bis 65535 als Namen einer Codepage, auch wenn er nicht die Standardform hat.

Beachten Sie, dass der Parameter HOSTCODEPAGE eine EBCDIC-basierte Codepage angeben muss, wenn eine Symbolverarbeitung erforderlich ist, da hier davon ausgegangen wird, dass die Begrenzungszeichen, die für die Verarbeitung von Symbolen und Symbollisten verwendet werden, in EBCDIC codiert sind.

LENGTH(datenwert)

Gibt die Länge des Puffer als Vollwort-Binärwert an, der die Daten für die Optionen TEXT, BINARY oder FROM enthält.

LISTLENGTH(datenwert)

Gibt die Länge der Symbolliste als Vollwort-Binärwert an.

SYMBOLLIST(datenbereich)

Gibt einen Puffer an, der eine Symbolliste enthält. Verwenden Sie die Option LISTLENGTH zur Angabe der Länge dieses Puffers. Eine Symbolliste ist eine Zeichenfolge, die aus einer oder mehreren Symboldefinitionen besteht, die durch Et-Zeichen getrennt angegeben sind. Jede Symboldefinition besteht aus einem Namen, einem Gleichheitszeichen und einem Wert. Das folgende Beispiel zeigt eine Symbolliste:

```
applid=IYCQ&jobname=test
```

Symbole in der Symbolliste werden standardmäßig durch das Et-Zeichen (&) getrennt. Sie können dies jedoch ändern, indem Sie mithilfe des Schlüsselworts DELIMITER ein anderes Symboltrennzeichen angeben. In [Regeln zur Angabe von Symbolen und Symbollisten](#) finden Sie eine Liste der Regeln, die beim Festlegen von Symbolen durch eine Symbolliste (SYMBOLLIST) gelten.

TEMPLATE(name)

Gibt einen 48 Byte langen Namen einer Vorlage an. Die Vorlage muss in CICS mit RDO definiert sein. Wenn der Name kürzer als 48 Byte ist, muss er auf der rechten Seite mit Leerzeichen aufgefüllt werden.

Anmerkung: Wenn Sie eine Vorlage einfügen, bevor die Symbole, die in ihr enthalten sind, festgelegt werden, werden die Symbole nie ersetzt. Dies kann geschehen, wenn Sie ein Dokument aus einer Vorlage erstellen, ohne eine Symbolliste anzugeben.

TEXT(datenbereich)

Gibt einen Puffer mit Daten an, die als Inhalt des neu zu erstellenden Dokuments verwendet werden sollen. Die Daten werden unverändert in den Dokumentinhalt kopiert. Es erfolgt kein Versuch, die Daten für die Symbolsubstitution zu parsen. Die Daten werden beim Senden des Dokuments mit einer Markierung versehen, die besagt, dass sie in die Client-Codepage konvertiert werden müssen. Verwenden Sie die Option LENGTH zur Angabe der Länge dieses Puffers.

UNESCAPED

Verhindert, dass CICS die Escapezeichen von den Symbolwerten entfernt, die im Puffer SYMBOLLIST enthalten sind. Wenn diese Option verwendet wird, werden Pluszeichen nicht in Leerzeichen konvertiert und Sequenzen wie %2B werden nicht in ein Byte lange Werte konvertiert.

In die Option UNESCAPED können Sie nicht das Zeichen einschließen, das Sie als Symboltrennzeichen in einem Symbolwert in einer Symbolliste verwendet haben. Wenn Sie die Option UNESCAPED verwenden wollen, wählen Sie ein Symboltrennzeichen aus, das nie in einem Symbolwert verwendet wird.

Bedingungen

INVREQ

RESP2-Wert:

1

Das abgerufene Dokument, das in der Option FROM angegeben wurde, hat kein gültiges RETRIEVE-Format.

LENGERR

RESP2-Wert:

1

Der für die Option LENGTH angegebene Wert ist ungültig. Der Wert ist negativ.

9

Der für die Option LISTLENGTH angegebene Wert ist ungültig. Der Wert muss zwischen 1 und (16 M - 1) liegen.

70 NOTAUTH

Eine Ressourcensicherheitsprüfung für den Befehl ist fehlgeschlagen. (Wenn die Bedingung NOTAUTH nicht behandelt wird, können Anwendungen, die diese Bedingung empfangen, mit dem Code AEY7 abnormal beendet werden.)

Beachten Sie, dass die Befehle **EXEC CICS DOCUMENT** auf Dokumentvorlagen durch den 48 Zeichen langen Namen der Vorlage verweisen (wie im Attribut TEMPLATENAME der DOCTEMPLATE-Ressource angegeben). Die Sicherheitsprüfung für die Befehle verwendet jedoch den Namen der DOCTEMPLATE-Ressourcendefinition, der dem Attribut TEMPLATENAME entspricht. Wenn die Ressourcensicherheitsprüfung eingerichtet ist, muss die Benutzer-ID für die Transaktion Lesezugriff (READ) auf diese DOCTEMPLATE-Ressourcendefinition haben.

RESP2-Wert:

101

Die Benutzer-ID für die Transaktion hat keinen Lesezugriff auf die DOCTEMPLATE-Ressourcendefinition für die Dokumentvorlage, die in der Option TEMPLATE angegeben wurde.

13 NOTFND

RESP2-Werte:

2

Das in der Option FROMDOC angegebene Dokument konnte nicht gefunden werden oder wurde falsch benannt.

3

Die in der Option TEMPLATE angegebene Vorlage konnte nicht gefunden werden oder wurde falsch benannt.

7

Die in der Option HOSTCODEPAGE angegebene Host-Codepage konnte nicht gefunden werden oder wurde falsch benannt.

8

Der für DELIMITER angegebene Wert ist nicht gültig.

116 SYMBOLERR

Ein in der Symbolliste angegebenes Symbol entspricht nicht den Benennungsregeln für Symbole. RESP2 enthält die relative Position (Offset) des Symbols in der Liste.

117 TEMPLATERR

Die in der DOCTEMPLATE-Ressource angegebene Vorlage ist nicht vorhanden oder es wurde ein ungültiger Befehl #set, #include oder #echo bei der Verarbeitung der angegebenen Vorlagendaten festgestellt. RESP2 enthält die relative Position (Offset) des ungültigen Befehls.

DOCUMENT DELETE

Löscht ein Dokument.

DOCUMENT DELETE

➡ DOCUMENT — DELETE — DOCTOKEN(*datenbereich*) —⬅

Bedingungen: NOTFND

Dieser Befehl ist threadsicher.

Beschreibung

Mit dem Befehl **DOCUMENT DELETE** können Sie Dokumente löschen, die während einer Transaktion nicht mehr benötigt werden. Durch den Befehl kann die Anwendung die Löschung eines Dokuments und sämtlichen Speichers, der zu dem Dokument gehört, anfordern. Nach der Ausführung dieses Befehls wird der Speicher, der dem Dokument zugeordnet ist, unverzüglich freigegeben. Wenn der Befehl **DOCUMENT DELETE** nicht aufgerufen wird, bleibt das Dokument so lange bestehen, bis die Anwendung beendet wird.

Optionen

DOCTOKEN(datenbereich)

Gibt das 16 Byte lange binäre Token des zu löschenden Dokuments an.

Bedingungen

13 NOTFND

RESP2-Werte:

1

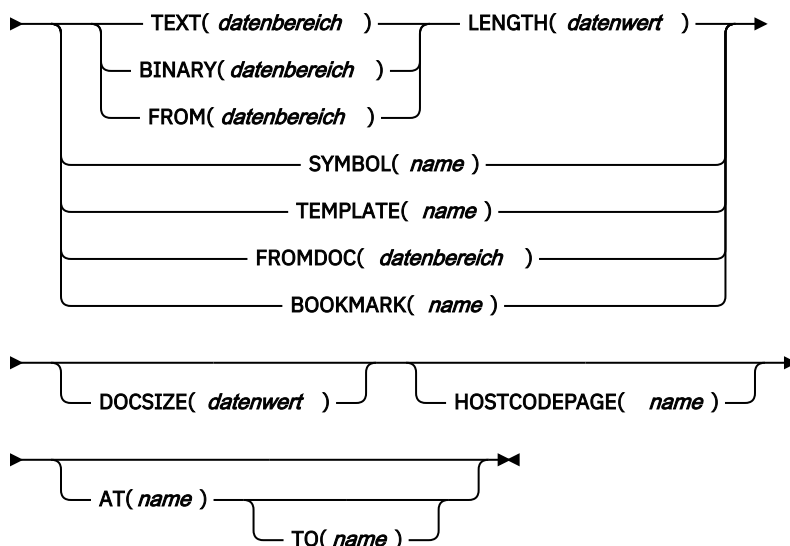
Das Dokument wurde nicht erstellt oder der Name wurde falsch angegeben.

DOCUMENT INSERT

Fügt Dokumentobjekte ein.

DOCUMENT INSERT

➡ DOCUMENT — INSERT — DOCTOKEN(*datenbereich*) →



Bedingungen: DUPREC, INVREQ, LENGERR, NOTAUTH, NOTFND, TEMPLATERR

Dieser Befehl ist threadsicher.

Beschreibung

Mit dem Befehl **DOCUMENT INSERT** kann die Anwendung Dokumentobjekte an Einfügepunkten in das Dokument einfügen. Die Einfügepunkte (Lesezeichen) definieren relative Positionen im Dokument. Lesezeichen müssen definiert werden, bevor auf sie verwiesen wird. Daten werden immer nach der Position eingefügt, die durch das Lesezeichen angegeben wird.

Optionen

AT(name)

Gibt den 16 Byte langen symbolischen Namen eines Lesezeichens an, das die Position des Einfügepunkts im Dokument identifiziert. Daten werden nach dem Lesezeichen eingefügt und alle Daten nach dem Lesezeichen werden nach unten verschoben. Die Anwendung kann eine Kombination der Optionen **AT** und **TO** verwenden, um eine Überlagerungsoperation durchzuführen. Wenn der Operand **AT** nicht angegeben wird, werden die Daten am Ende des Dokuments eingefügt. Es wird ein vordefinierter Lesezeichen **TOP** bereitgestellt, mit dessen Hilfe die Anwendung Daten am Anfang des Dokuments einfügen kann.

BINARY(datenbereich)

Gibt einen Puffer von Daten an, die in das Dokument eingefügt werden sollen. Die Daten werden unverändert an den Einfügepunkt im Dokument kopiert. Es erfolgt kein Versuch, die Daten für die Symbolsubstitution zu parsen. Mit der Option **BINARY** kann die Anwendung Blöcke von Daten einfügen, die keiner Konvertierung in eine Client-Codepage unterzogen werden dürfen, wenn die Daten gesendet werden. Verwenden Sie die Option **LENGTH** zur Angabe der Länge dieses Puffers.

BOOKMARK(name)

Gibt ein Lesezeichen an, das in das Dokument eingefügt werden soll. Ein Lesezeichen ist ein symbolischer Name, der einen Einfügepunkt im Dokument identifiziert. Der Name kann bis zu 16 Zeichen lang sein und darf keine eingebetteten Leerzeichen enthalten.

DOCSIZE(datenwert)

Gibt einen binären Vollwortbereich an, der mit der aktuellen Größe des Dokuments in Byte aktualisiert werden soll. Dies ist die maximale Größe des Puffers, der für eine Kopie des Dokuments benötigt wird, wenn ein Befehl RETRIEVE abgesetzt wird.

DOCTOKEN(datenbereich)

Gibt das 16 Byte lange binäre Token des Dokuments an, in das die Daten eingefügt werden sollen.

FROM(datenbereich)

Gibt an, dass ein Puffer von Daten, die von der Anwendung bereitgestellt werden, die in das Dokument eingefügt werden soll. Der Dateninhalt kann eine Vorlage oder ein Dokument sein, die bzw. das zuvor erstellt und abgerufen wurde. Wenn die Daten eine Vorlage sind, erfolgt eine Symbolsubstitution dort, wo Symbole in der Symboltabelle vorhanden sind. Wenn die Daten ein zuvor abgerufenes Dokument sind, werden die Konvertierungs- und Lesezeichentags, die beim Abruf eingefügt wurden, aus dem Inhalt entfernt und in dem internen Format gespeichert, das für die API-Befehle erforderlich ist. Beachten Sie, dass keine Symbolsubstitution von nicht aufgelösten Symbolen erfolgt, die in einem abgerufenen Dokument enthalten sind. Verwenden Sie die Option LENGTH zur Angabe der Länge dieses Puffers.

FROMDOC(datenbereich)

Gibt das binäre Token eines Dokuments an (siehe Option DOCTOKEN), dessen Inhalt an den Einfügepunkt des Zieldokuments kopiert wird. Alle Lesezeichen und Konvertierungstags werden in das neue Dokument kopiert. Die Symboltabelle wird nicht kopiert.

HOSTCODEPAGE(name)

Gibt den symbolischen Namen (siehe Option DOCTOKEN) der Host-Codepage an, in der die Daten, die hinzugefügt werden, codiert werden. Diese Option gilt nur für die Optionen TEXT, SYMBOL und TEMPLATE. Der Name muss acht Zeichen lang sein. Ist er kürzer als acht Zeichen, muss er auf der rechten Seite mit Leerzeichen aufgefüllt werden.

Das CICS-Standardformat für einen Codepagenamen besteht aus der Codepagenummer (oder allgemeiner der CCSID), die nach Bedarf mithilfe von 3 bis 5 Dezimalziffern geschrieben und anschließend mit nachgestellten Leerzeichen auf 8 Zeichen aufgefüllt wird. Bei der Codepage 37, die aus weniger als 3 Ziffern besteht, lautet das Standardformat '037'. CICS akzeptiert jetzt jede Dezimalzahl aus bis zu acht Ziffern (mit nachfolgenden Leerzeichen aufgefüllt) aus dem Bereich von 1 bis 65535 als Namen einer Codepage, auch wenn er nicht die Standardform hat.

Beachten Sie, dass der Parameter HOSTCODEPAGE eine EBCDIC-basierte Codepage angeben muss, wenn eine Symbolverarbeitung erforderlich ist, da hier davon ausgegangen wird, dass die Begrenzungszeichen, die für die Verarbeitung von Symbolen und Symbollisten verwendet werden, in EBCDIC codiert sind.

LENGTH(datenwert)

Gibt die Länge des Puffers als Vollwort-Binärwert an, der die Daten für die Optionen TEXT, BINARY oder FROM enthält.

Wenn der Befehl DOCUMENT INSERT auf einen Befehl DOCUMENT RETRIEVE ohne Angabe der Option DATAONLY folgt und das abgerufene Dokument mithilfe der Option FROM eingefügt werden soll, muss die in LENGTH angegebene Länge mit der Länge des abgerufenen Dokuments übereinstimmen.

SYMBOL(name)

Gibt den 32 Byte langen Namen eines Symbols in der Symboltabelle an. Die Daten, die dem Symbol in der Symboltabelle zugeordnet sind, werden eingefügt, nicht jedoch das Symbol selbst. Beachten Sie, dass Sie, wenn Daten, die einem Symbol zugeordnet sind, in das Dokument eingefügt wurden, diese Daten in dem Dokument, das gerade zusammengesetzt wird, nicht ändern können. Wenn Sie einen Wert für das Symbol festlegen, wird der neue Wert bei der nächsten Einfügung dieses Symbols in ein Dokument verwendet. Ihre Änderung hat keine Auswirkung auf den Wert, der bereits in das Dokument eingefügt wurde.

TEMPLATE(name)

Gibt einen 48 Byte langen Namen einer Vorlage an. Die Vorlage muss in CICS mit RDO definiert sein. Wenn der Name kürzer als 48 Byte ist, muss er auf der rechten Seite mit Leerzeichen aufgefüllt werden. Die aktuellen Werte von Symbolen werden in die Vorlage eingesetzt.

Anmerkung: Wenn eine Vorlage, die Symbole enthält, in ein Dokument eingefügt wurde, können Sie die eingesetzten Werte dieser Symbole in dem Dokument, das gerade zusammengesetzt wird, nicht ändern. Wenn Sie andere Werte für die Symbole festlegen, werden die neuen Werte bei der nächsten Einfügung der Vorlage in ein Dokument verwendet. Ihre Änderungen haben keine Auswirkungen auf die Werte, die bereits in das Dokument eingefügt wurden.

TEXT(datenbereich)

Gibt einen Puffer von Daten an, die in das Dokument eingefügt werden sollen. Die Daten werden unverändert an den Einfügpunkt im Dokument kopiert. Es erfolgt kein Versuch, die Daten für die Symbolsubstitution zu parsen. Wenn das Dokument gesendet wird, wird es mit einer Markierung versehen, die besagt, dass es in die Client-Codepage konvertiert werden muss. Verwenden Sie die Option LENGTH zur Angabe der Länge dieses Puffers.

TO(name)

Gibt den symbolischen Namen eines Lesezeichens an, das die Endposition einer Überlagerungsoperation identifiziert. Daten zwischen Lesezeichen, die durch die Operanden AT und TO angegeben werden, werden gelöscht und neue Daten an ihrer Stelle eingefügt. Es ist möglich, Daten zwischen zwei Lesezeichen zu löschen, indem eine Nullzeichenfolge in der Option TEXT oder BINARY mit der Länge (LENGTH) null angegeben wird.

Bedingungen

14 DUPREC

Das Lesezeichen wurde bereits definiert.

16 INVREQ

Die RESP2-Werte lauten wie folgt:

0

Das Lesezeichen, das in der Option TO angegeben wird, steht vor dem Lesezeichen, das mit der Option AT angegeben wird.

1

Das abgerufene Dokument, das in der Option FROM angegeben wurde, hat kein gültiges RETRIEVE-Format.

2

Der Lesezeichenname in der Option BOOKMARK ist ungültig.

LENGERR

RESP2-Wert:

1

Der für die Option LENGTH angegebene Wert ist ungültig. Der Wert ist negativ.

70 NOTAUTH

Eine Ressourcensicherheitsprüfung für den Befehl ist fehlgeschlagen. (Wenn die Bedingung NOTAUTH nicht behandelt wird, können Anwendungen, die diese Bedingung empfangen, mit dem Code AEY7 abnormal beendet werden.)

Beachten Sie, dass die Befehle EXEC CICS DOCUMENT auf Dokumentvorlagen durch den 48 Zeichen langen Namen der Vorlage verweisen (wie im Attribut TEMPLATENAME der DOCTEMPLATE-Ressource angegeben). Die Sicherheitsprüfung für die Befehle verwendet jedoch den Namen der DOCTEMPLATE-Ressourcendefinition, der dem Attribut TEMPLATENAME entspricht. Wenn die Ressourcensicherheitsprüfung eingerichtet ist, muss die Benutzer-ID für die Transaktion Lesezugriff (READ) auf diese DOCTEMPLATE-Ressourcendefinition haben.

RESP2-Wert:

101

Die Benutzer-ID für die Transaktion hat keinen Lesezugriff auf die DOCTEMPLATE-Ressourcendefinition für die Dokumentvorlage, die in der Option TEMPLATE angegeben wurde.

13 NOTFND

Eines der folgenden Dokumente bzw. eine der folgenden Vorlagen konnte nicht gefunden werden oder der Name wurde falsch angegeben:

RESP2-Werte:

- 1**
Das in der Option DOCUMENT angegebene Dokument.
- 2**
Das in der Option FROMDOC angegebene Dokument.
- 3**
Die in der Option TEMPLATE angegebene Vorlage.
- 4**
Das in der Option SYMBOL angegebene Dokument.
- 5**
Das in der Option AT angegebene Dokument.
- 6**
Das in der Option TO angegebene Dokument.
- 7**
Das in der Option HOSTCODEPAGE angegebene Dokument.

117 TEMPLATERR

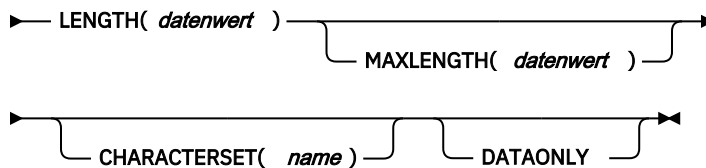
Bei der Verarbeitung der angegebenen Vorlagendaten wurde ein ungültiger Befehl #set, #include oder #echo festgestellt oder die CICS-Region verfügt nicht über die richtige Berechtigungsebene für den Zugriff auf die zFS-Datei dieser Vorlage. RESP2 enthält entweder null (wenn das Maximum von 32 Ebenen für eingebettete Vorlagen überschritten wurde) oder die relative Position (Offset) des ungültigen Befehls.

DOCUMENT RETRIEVE

Kopiert ein Dokument in den Puffer der Anwendung.

DOCUMENT RETRIEVE

➤ DOCUMENT — RETRIEVE — DOCTOKEN(*datenbereich*) — INTO(*datenbereich*) ➤



Bedingungen: INVREQ, LENGERR, NOTFND

Dieser Befehl ist threadsicher.

Beschreibung

Mit dem Befehl **DOCUMENT RETRIEVE** kann die Anwendung eine Kopie des Dokuments in den eigenen Puffer abrufen, der anschließend direkt bearbeitet werden kann. Das Dokument wird von CICS verwaltet und die Anwendung hat keinen direkten Zugriff auf den Puffer, der den Inhalt des Dokuments enthält. Das Dokument ist nur für die Dauer der aktuellen Transaktion vorhanden, sodass die Anwendung das Dokument abrufen und speichern muss, wenn das Dokument über Transaktionsgrenzen hinweg vorhanden sein soll. Das abgerufene Dokument kann als Basis für ein neues Dokument mithilfe der Option FROM des Befehls **DOCUMENT CREATE** verwendet werden.

Wenn das Dokument abgerufen wurde, fügt CICS Tags in den Dokumentinhalt ein, um die Lesezeichen zu identifizieren und die Blöcke abzugrenzen, für die keine Codepagekonvertierung erforderlich ist. Zur An-

forderung einer Kopie ohne Tags können Sie die Option DATAONLY angeben. Das extrahierte Dokument kann mithilfe der Option CHARACTERSET auch in eine einzelne Client-Codepage konvertiert werden.

Die Befehle **DOCUMENT CREATE** und **DOCUMENT INSERT** geben einen DOCSIZE-Wert zurück. Dieser Wert gibt die maximale Größe des Puffers an, der zur Aufnahme einer Kopie des Dokuments in seiner ursprünglichen Codepage (einschließlich Steuerinformationen) benötigt wird, wenn der Befehl RETRIEVE abgesetzt wird. Wenn die Option CHARACTERSET jedoch eine Codierung angibt, die mehr Byte als die ursprünglichen EBCDIC-Daten erfordert (z. B. UTF-8), reicht die maximale Größe möglicherweise nicht aus, um das konvertierte Dokument zu speichern. Wenn der DOCSIZE-Wert in diesem Fall für die Puffergröße verwendet wird, sollte das Programm darauf vorbereitet sein, eine Bedingung LENGERR zu behandeln und einen neuen Puffer anzufordern, der die Größe hat, die im Parameter LENGTH zurückgegeben wird. Alternativ können Sie die tatsächliche Dokumentlänge ermitteln, bevor Sie den Puffer zuordnen, indem Sie einen Befehl **DOCUMENT RETRIEVE** mit einem Pseudo-Puffer und dem Wert null für MAXLENGTH absetzen und anschließend die Bedingung LENGERR unter Verwendung des zurückgegebenen Werts für LENGTH verarbeiten.

Optionen

CHARACTERSET(name)

Gibt den Namen des Client-Zeichensatzes an, in den die Daten konvertiert werden sollen. Der Name kann bis zu 40 Zeichen lang sein. Ist er kürzer als 40 Zeichen, muss er auf der rechten Seite mit Leerzeichen aufgefüllt werden.

Dieser Parameter ersetzt den Parameter CLNTCODEPAGE, der nur zu Upgradezwecken unterstützt wird.

CLNTCODEPAGE(name)

Diese Option wird nur zu Upgradezwecken unterstützt. Sie wurde durch die Option CHARACTERSET ersetzt. Bei beiden Schlüsselwörtern führt CICS dieselbe Aktion aus.

DATAONLY

Gibt an, dass die Daten ganz ohne eingebettete Tags abgerufen werden sollen.

DOCTOKEN(datenbereich)

Gibt das 16 Byte lange binäre Token des abzurufenden Dokuments an.

INTO(datenbereich)

Gibt den Puffer an, der die Kopie des Dokumentinhalts aufnehmen soll.

LENGTH(datenwert)

Gibt die Länge der Daten, die an die Anwendung zurückgegeben werden, als Vollwort-Binärwert an. Wenn das Dokument abgeschnitten wird, ist dies die genaue Länge, die zur Rückgabe des gesamten Dokuments erforderlich ist.

MAXLENGTH(datenwert)

Gibt die Länge der maximalen Datenmenge, die der Puffer empfangen kann, als Vollwort-Binärwert an.

Bedingungen

16 INVREQ

RESP2-Werte:

11

Eine ungültige oder nicht unterstützte Kombination von Codepages wurde angegeben.

12

Bei der CCSID-Konvertierung ist ein Fehler aufgetreten und die Konvertierung konnte nicht ausgeführt werden. Beispiel: Ein Textstück oder ein Symbolwert endet nach einem Teil des gesamten Mehrbytezeichens.

22 LENGERR

RESP2-Werte:

1

MAXLENGTH ist kleiner null. Die Dokumentgröße wird nicht im Feld LENGTH zurückgegeben.

2

Die Länge des empfangenden Puffers ist null oder zu kurz, um den Dokumentinhalt aufzunehmen. Das Dokument wurde abgeschnitten und die genaue erforderliche Länge wird im Feld LENGTH zurückgegeben.

13 NOTFND

RESP2-Werte:

1

Das Dokument wurde nicht erstellt oder der Name wurde falsch angegeben.

7

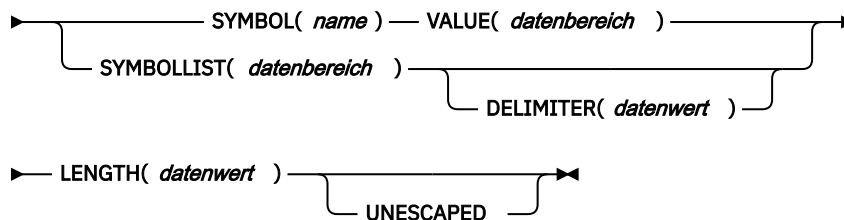
Der angegebene Zeichensatz kann nicht gefunden werden.

DOCUMENT SET

Fügt Symbole und Werte zur Symboltabelle hinzu.

DOCUMENT SET

➔ DOCUMENT — SET — DOCTOKEN(*datenbereich*) ➔



Bedingungen: INVREQ, LENGERR, NOTFND, SYMBOLERR

Dieser Befehl ist threadsicher.

Beschreibung

Mit dem Befehl DOCUMENT SET kann die Anwendung der Symboltabelle Symbole und deren zugeordnete Werte hinzufügen. Wenn das Symbol, das hinzugefügt werden sollen, bereits in der Tabelle vorhanden ist, wird es durch die neue Definition ersetzt.

Anmerkung:

1. Wenn eine Vorlage, die Symbole enthält, in ein Dokument eingefügt wurde, können Sie die eingesetzten Werte dieser Symbole in dem Dokument, das gerade zusammengesetzt wird, nicht ändern. Wenn Sie andere Werte für die Symbole festlegen, werden die neuen Werte bei der nächsten Einfügung der Vorlage in ein Dokument verwendet. Ihre Änderungen haben keine Auswirkungen auf die Werte, die bereits in das Dokument eingefügt wurden.
2. Wenn Sie eine Vorlage einfügen, bevor die Symbole, die in ihr enthalten sind, festgelegt werden, werden die Symbole nie ersetzt. Dies kann geschehen, wenn Sie ein Dokument aus einer Vorlage erstellen, ohne eine Sybolliste anzugeben.

Optionen

DELIMITER(datenwert)

Gibt einen optionalen 1-Byte-Wert an, mit dem Symbolname/Wert-Paare im Puffer SYMBOLLIST begrenzt werden. Wenn diese Option nicht angegeben wird, wird standardmäßig ein Et-Zeichen verwendet. Bestimmte Begrenzungszeichen (z. B. das Leerzeichen) sind nicht zulässig und verursachen bei Verwendung sämtlich eine Bedingung INVREQ für den Befehl. Sie werden unter [Regeln zur Angabe von Symbolen und Sybollisten](#) aufgelistet.

Bei Verwendung dieser Option muss die Anwendung sicherstellen, dass das Begrenzungszeichen (DELIMITER) nicht als Symbolwert im Puffer SYMBOLLIST vorkommt. Aus diesem Grund sollte die Anwendung keine alphanumerischen oder anderen druckbaren Zeichen als Wert für DELIMITER verwenden.

DOCTOKEN(*datenbereich*)

Gibt das 16 Byte lange binäre Token des Dokuments an, zu dem die Symboltabelle gehört.

LENGTH(*datenwert*)

Gibt die Länge des Puffers mit dem den Datenwert, der dem Symbol zugeordnet ist, oder mit der Länge des Puffers, der bei Verwendung der Option SYMBOLLIST die Symbolliste enthält, als Vollwort-Binärwert an.

SYMBOL(*name*)

Gibt den Namen des Symbols an, das der Tabelle hinzugefügt werden soll. Der Name kann 1 bis 32 Zeichen lang sein und keine eingebetteten Leerzeichen enthalten. In [Regeln zur Angabe von Symbolen und Symbollisten](#) finden Sie eine Liste der Regeln, die für die Angabe des Namens eines Symbols gelten. Wenn mehr als ein Symbol in demselben Befehl definieren wollen, verwenden Sie stattdessen die Option SYMBOLLIST.

SYMBOLLIST(*datenbereich*)

Gibt einen Puffer an, der eine Symbolliste enthält. Verwenden Sie die Option LENGTH zur Angabe der Länge dieses Puffers. Eine Symbolliste ist eine Zeichenfolge, die aus einer oder mehreren Symboldefinitionen besteht, die durch Et-Zeichen getrennt angegeben sind. Jede Symboldefinition besteht aus einem Namen, einem Gleichheitszeichen und einem Wert. Das folgende Beispiel zeigt eine Symbolliste:

```
applid=IYCQ&jobname=test
```

Symbole in der Symbolliste werden standardmäßig durch das Et-Zeichen (&) getrennt. Sie können dies jedoch ändern, indem Sie mithilfe des Schlüsselworts DELIMITER ein anderes Symboltrennzeichen angeben. In [Regeln zur Angabe von Symbolen und Symbollisten](#) finden Sie eine Liste der Regeln, die beim Festlegen von Symbolen durch eine Symbolliste (SYMBOLLIST) gelten.

UNESCAPED

Verhindert, dass CICS die Escapezeichen von den Symbolwerten entfernt, die im Puffer SYMBOLLIST enthalten sind. Wenn diese Option verwendet wird, werden Pluszeichen nicht in Leerzeichen konvertiert und Sequenzen wie %2B werden nicht in ein Byte lange Werte konvertiert.

In die Option UNESCAPED können Sie nicht das Zeichen einschließen, das Sie als Symboltrennzeichen in einem Symbolwert in einer Symbolliste verwendet haben. Wenn Sie die Option UNESCAPED verwenden wollen, wählen Sie ein Symboltrennzeichen aus, das nie in einem Symbolwert verwendet wird. Alternativ können Sie die Optionen SYMBOL und VALUE verwenden, um Symbolwerte anzugeben, die das Zeichen enthalten, das Sie als Symboltrennzeichen verwendet haben, weil das Symboltrennzeichen keine besondere Bedeutung hat, wenn es in der Option VALUE verwendet wird.

VALUE(*datenbereich*)

Gibt einen Bereich an, der den Wert enthält, der dem Symbol (SYMBOL) zugeordnet werden soll. In [Regeln zur Angabe von Symbolen und Symbollisten](#) finden Sie eine Liste der Regeln, die für die Angabe des Werts eines Symbols gelten.

Bedingungen

16 INVREQ

RESP2-Werte:

8

Der für DELIMITER angegebene Wert ist nicht gültig.

LENGERR

RESP2-Wert:

9

Der für die Option LENGTH der Symbolliste angegebene Wert ist ungültig. Der Wert muss zwischen 1 und (16 M - 1) liegen.

10

Der für die Option LENGTH des Symbolwerts angegebene Wert ist ungültig. Der Wert muss zwischen 1 und (16 M - 1) liegen.

13 NOTFND

RESP2-Werte:

1

Das Dokument wurde nicht erstellt oder der Name wurde falsch angegeben.

116 SYMBOLERR

Ein Symbolname ist ungültig. RESP2-Werte:

0

SYMBOLLIST wurde nicht verwendet.

offset

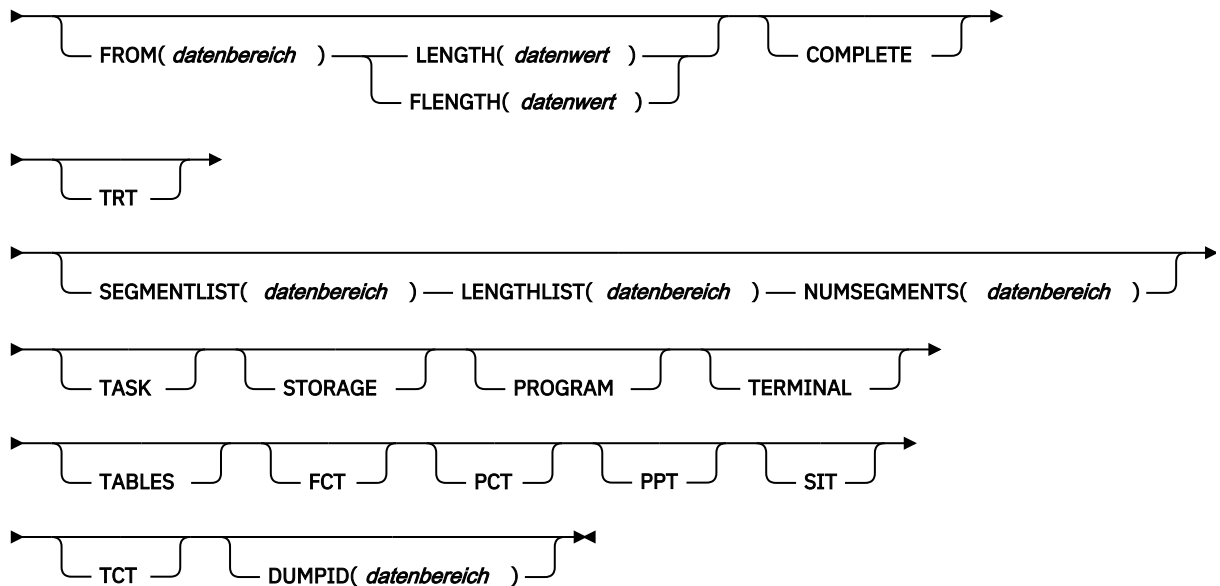
RESP2 enthält die relative Position (Offset) des ungültigen Symbols in der Liste.

DUMP TRANSACTION

Fordert einen Transaktionsspeicherauszug an.

DUMP TRANSACTION

►► DUMP TRANSACTION — DUMPCODE(*name*) →



Bedingungen: INVREQ, IOERR, NOSPACE, NOSTG, NOTOPEN, OPENERR, SUPPRESSED

Beschreibung

Der Befehl DUMP TRANSACTION generiert einen Speicherauszug aller Hauptspeicherbereiche, einer Reihe von Hauptspeicherbereichen oder eines einzelnen Hauptspeicherbereichs, die bzw. der zu einer Task gehört, sowie bestimmter oder aller CICS-Tabellen oder aller dieser Elemente zusammen.

Beachten Sie, dass Sie, wenn Sie einen Befehl DUMP TRANSACTION für einen Speicherauszugscode (DUMPCODE) absetzen, der in der Transaktionsspeicherauszugstabelle mit SYSDUMP definiert ist, auch einen Systemspeicherauszug erhalten.

Falls kein Eintrag in der Systemspeicherauszugstabelle für den angegebenen Speicherauszugscode vorhanden ist, wird ein temporärer Eintrag erstellt. Dieser Eintrag geht beim nächsten Start von CICS verloren. Die Systemspeicherauszugstabelle wird in Speicherauszugscodes, die Sie angeben können beschrieben.

Optionen

COMPLETE

Erstellt einen Speicherauszug aller Hauptspeicherbereiche, die zu einer Task gehören, aller CICS-Tabellen sowie der DL/I-Steuerblöcke.

DUMPCODE(name)

Gibt einen Namen (1-4 Zeichen) an, der den Speicherauszug identifiziert. Wenn der Name führende oder eingebettete Leerzeichen enthält, wird der Speicherauszug generiert, jedoch eine Bedingung INVREQ ausgelöst. Der Systemspeicherauszugstabelle wird kein Eintrag hinzugefügt.

Wenn Sie alle Optionen außer DUMPCODE weglassen, erhalten Sie den gleichen Speicherauszug wie bei Angabe von TASK, jedoch ohne die DL/I-Steuerblöcke.

DUMPID(datenbereich)

Gibt eine Speicherauszugs-ID aus 6 - 9 Zeichen zurück, die für diesen speziellen Speicherauszug generiert wird. Das Format der ID ist xxxx/yyyy, wobei xxxx die **Ausführungsnummer des Speicherauszugs** darstellt, yyyy den **Speicherauszugszähler** darstellt und das Schrägstrichsymbol (/) ein Trennzeichen darstellt. Die Speicherauszugs-ID wird wie folgt generiert:

Ausführungsnummer des Speicherauszugs

Eine Zahl aus dem Bereich von 1 bis 9999. (Führende Nullen werden für diese Nummer nicht verwendet, sodass die Speicherauszugs-ID in der Länge zwischen 6 und 9 Zeichen variieren kann.) Die Ausführungsnummer des Speicherauszugs beginnt bei 1, wenn Sie CICS zu Anfang mit einem neu initialisierten lokalen Katalog starten, und wird jeweils um 1 erhöht, wenn Sie CICS erneut starten. Die Ausführungsnummer des Speicherauszugs wird im lokalen Katalog gespeichert, wenn Sie das System normal herunter fahren. Es wird jedoch zurückgesetzt, wenn Sie CICS mit dem Systeminitialisierungsparameter START=INITIAL oder START=COLD erneut starten.

Speicherauszugszähler

Eine Zahl aus dem Bereich von 0001 bis 9999. (Führende Nullen sind in der Speicherauszugs-ID erforderlich.) Dieser Wert wird dem Speicherauszug in dieser Ausführung von CICS, beginnend mit 0001 für den ersten Speicherauszug und anschließend mit jedem erfassten Speicherauszug um 1 erhöht, zugeordnet.

FCT

Erstellt einen Speicherauszug der Dateisteuertabelle, die FILE-Ressourcendefinitionen enthält.

FLENGTH(datenwert)

Gibt die Länge (Vollwort-Binärwert) des Speicherbereichs (der in der Option FROM angegeben wird) an, von dem ein Speicherauszug erstellt werden soll. Die maximale Länge, die Sie angeben können, beträgt 16 777 215 Byte.

Die Optionen FLENGTH und LENGTH schließen sich gegenseitig aus.

FROM(datenbereich)

Erstellt einen Speicherauszug des angegebenen Datenbereichs, bei dem es sich um einen gültigen Bereich handeln muss. Das heißt, es muss Speicher sein, der vom Betriebssystem innerhalb der CICS-Region zugeordnet wurde. Darüber hinaus wird ein Speicherauszug der folgenden Bereiche erstellt:

- Tasksteuerbereich (TCA) und, sofern zutreffend, Transaktionsarbeitsbereich (TWA).
- Allgemeiner Systembereich (CSA), einschließlich des Benutzerteils des CSA (CWA).
- Wenn TRAN für den SIT-Parameter TRTRANTY angegeben ist, werden nur die Traceeinträge formatiert, die der aktuellen Task zugeordnet sind. Wenn TRTRANTY=ALL angegeben ist, wird die gesamte interne Tracetabelle formatiert. Dies gilt nur, wenn die CICS-Tracefunktion aktiv ist.
- Entweder der Terminaleintrag in der Terminalsteuertabelle (TCTTE) oder die Definition der Warteschlange mit transienten Daten mit der anfordernden Task.

Immer wenn ein Speicherauszug des TCTTE erstellt wird, wird ein Speicherauszug des Benutzerbereichs der Terminalsteuertabelle (sofern vorhanden) und der Nachrichtensteuerblöcke (sofern vorhanden), die dem TCTTE zugeordnet sind, erstellt. Die letzteren werden von Basic Mapping Support (BMS) verwendet.

LENGTH(datenwert)

Gibt die Länge (Halbwort-Binärwert) des Datenbereichs an, der in der Option FROM angegeben wird. Eine Beschreibung einer sicheren Obergrenze finden Sie unter „[LENGTH-Optionen in CICS-Befehlen](#)“ auf Seite 10.

Die Optionen LENGTH und FLENGTH schließen sich gegenseitig aus.

LENGTHLIST(datenbereich)

Gibt eine Liste von 32-Bit-Binärwerten an, die die Längen der Speicherbereiche zeigen, für die ein Speicherauszug erstellt werden soll. Diese Liste entspricht der Liste von Segmenten, die in der Option SEGMENTLIST angegeben wird. Sie müssen die beiden Optionen SEGMENTLIST und NUMSEGMENTS verwenden, wenn Sie die Option LENGTHLIST verwenden.

NUMSEGMENTS(datenbereich)

Gibt die Anzahl (Vollwort-Binärwert) von Bereichen an, für die ein Speicherauszug erstellt werden soll. Sie müssen die beiden Optionen SEGMENTLIST und LENGTHLIST verwenden, wenn Sie die Option NUMSEGMENTS verwenden.

PCT

Formatiert eine Zusammenfassung jeder installierten Transaktionsressourcendefinition.

PPT

Formatiert eine Zusammenfassung jeder installierten Programmressourcendefinition.

PROGRAM

Gibt an, dass ein Speicherauszug für die folgenden Programmspeicherbereiche, die dieser Task zugeordnet sind, erstellt werden soll:

- Tasksteuerbereich (TCA) und, sofern zutreffend, Transaktionsarbeitsbereich (TWA).
- Allgemeiner Systembereich (CSA), einschließlich des Benutzerteils des CSA (CWA).
- Traceeinträge, die sich auf die Task beziehen und in die interne Tracetabelle geschrieben wurden (nur wenn die CICS-Tracefunktion aktiv ist).
- Alle Programmspeicherbereiche, die benutzergeschriebene Anwendungsprogramme enthalten, die für die anfordernde Task aktiv sind.
- Registersicherungsbereiche (RSAs), die von der RSA-Kette aus dem TCA angegeben werden.
- Entweder der Terminaleintrag in der Terminalsteuertabelle (TCTTE) oder die Definition der Warteschlange mit transienten Daten mit der anfordernden Task.

Immer wenn ein Speicherauszug des TCTTE erstellt wird, wird ein Speicherauszug des Benutzerbereichs der Terminalsteuertabelle (sofern vorhanden) und der Nachrichtensteuerblöcke (sofern vorhanden), die dem TCTTE zugeordnet sind, erstellt.

SEGMENTLIST(datenbereich)

Gibt eine Liste von Adressen an, die Ausgangspunkte der Segmente sind, für die ein Speicherauszug erstellt werden soll. Jedes Segment ist ein taskbezogener Speicherbereich. Sie müssen die beiden Optionen NUMSEGMENTS und LENGTHLIST verwenden, wenn Sie die Option SEGMENTLIST verwenden.

SIT

Erstellt einen Speicherauszug der Systeminitialisierungstabelle (SIT).

STORAGE

Gibt an, dass ein Speicherauszug für die folgenden Speicherbereiche, die dieser Task zugeordnet sind, erstellt werden soll:

- Tasksteuerbereich (TCA) und, sofern zutreffend, Transaktionsarbeitsbereich (TWA).
- Allgemeiner Systembereich (CSA), einschließlich des Benutzerteils des CSA (CWA).

- Traceeinträge, die sich auf die Task beziehen und in die interne Tracetabelle geschrieben wurden (nur wenn die CICS-Tracefunktion aktiv ist).
- Alle Transaktionsspeicherbereiche.
- Entweder der Terminaleintrag in der Terminalsteuertabelle (TCTTE) oder die Definition der Warteschlange mit transienten Daten mit der anfordernden Task.

Immer wenn ein Speicherauszug des TCTTE erstellt wird, wird ein Speicherauszug des Benutzerbereichs der Terminalsteuertabelle (sofern vorhanden) und der Nachrichtensteuerblöcke (sofern vorhanden), die dem TCTTE zugeordnet sind, erstellt.

TABLES

Erstellt einen Speicherauszug der Dateisteuertabelle (FCT), die die FILE-Ressourcendefinitionen enthält, der Programmsteuertabelle (PCT), die die TRANSACTION-Ressourcendefinitionen enthält, der Verarbeitungsprogrammtabelle (PPT), die die PROGRAM-Ressourcendefinitionen enthält, der Systeminitialisierungstabelle (SIT), die die CICS-Systeminitialisierungsparameter enthält, sowie der Terminalsteuertabelle (TCT), die die TERMINAL-Ressourcendefinitionen enthält.

TASK

Gibt an, dass ein Speicherauszug für die folgenden Speicherbereiche, die dieser Task zugeordnet sind, erstellt werden soll:

- Eine Zusammenfassung der Transaktionsumgebung, die dieser Task zugeordnet ist.
- Allgemeiner Systembereich (CSA), einschließlich des Benutzerteils des CSA (CWA).
- Traceeinträge, die sich auf die Task beziehen und in die interne Tracetabelle geschrieben wurden (nur wenn die CICS-Tracefunktion aktiv ist).
- Alle Programmspeicherbereiche, die benutzergeschriebene Anwendungsprogramme enthalten, die für die anfordernde Task aktiv sind.
- Alle Transaktionsspeicherbereiche.
- Entweder der Terminaleintrag in der Terminalsteuertabelle (TCTTE) oder die Definition der Warteschlange mit transienten Daten mit der anfordernden Task.
- Registersicherungsbereiche (RSAs), die von der RSA-Kette aus dem TCA angegeben werden.
- Alle Terminal-E/A-Bereiche (TIOAs), die aus dem Terminaleintrag der Terminalsteuertabelle (TCTTE) für das Terminal verkettet wurden, das der anfordernden Task zugeordnet ist.
- DL/I-Steuerblöcke.

Immer wenn ein Speicherauszug des TCTTE erstellt wird, wird ein Speicherauszug des Benutzerbereichs der Terminalsteuertabelle (sofern vorhanden) und der Nachrichtensteuerblöcke (sofern vorhanden), die dem TCTTE zugeordnet sind, erstellt.

TCT

Erstellt einen Speicherauszug der Terminalsteuertabelle (TCT).

TERMINAL

Gibt an, dass ein Speicherauszug für die folgenden Speicherbereiche, die dem Terminal zugeordnet sind, erstellt werden soll:

- Tasksteuerbereich (TCA) und, sofern zutreffend, Transaktionsarbeitsbereich (TWA).
- Allgemeiner Systembereich (CSA), einschließlich des Benutzerteils des CSA (CWA).
- Traceeinträge, die sich auf die Task beziehen und in die interne Tracetabelle geschrieben wurden (nur wenn die CICS-Tracefunktion aktiv ist).
- Alle Terminal-E/A-Bereiche (TIOAs), die aus dem Terminaleintrag der Terminalsteuertabelle (TCTTE) für das Terminal verkettet wurden, das der anfordernden Task zugeordnet ist, solange die Anforderung keine Schreibanforderung ist oder die Speicherblockierung für die Task oder das Terminal aktiv ist.
- Entweder der Terminaleintrag in der Terminalsteuertabelle (TCTTE) oder die Definition der Warteschlange mit transienten Daten mit der anfordernden Task.

Immer wenn ein Speicherauszug des TCTTE erstellt wird, wird ein Speicherauszug des Benutzerbereichs der Terminalsteuertabelle (sofern vorhanden) und der Nachrichtensteuerblöcke (sofern vorhanden), die dem TCTTE zugeordnet sind, erstellt. Die letzteren werden von Basic Mapping Support (BMS) verwendet.

TRT

Erstellt einen Speicherauszug der Traceeinträge in Bezug auf die Task, die in die interne Tracetabelle geschrieben wurden.

Bedingungen

16 INVREQ

RESP2-Werte:

13

Es wurde ein falscher DUMPCODE angegeben. DUMPCODE enthält nicht druckbare Zeichen oder führende oder eingebettete Leerzeichen.

Der Speicherauszug wird generiert, jedoch wird der Systemspeicherauszugstabelle kein Eintrag hinzugefügt.

Standardaktion: Die Task wird abnormal beendet.

17 IOERR

RESP2-Werte:

9

Der SDUMP-Prozess ist nicht autorisiert.

10

Während der Erstellung des Speicherauszugs ist ein Fehler aufgetreten.

13

Die CICS-Routine, die SDUMP absetzt, kann keine Wiederherstellungsroutine (FESTAE) einrichten.

Standardaktion: Die Task wird abnormal beendet.

18 NOSPACE

RESP2-Werte:

4

Der Transaktionsspeicherauszug ist aufgrund von Speichermangel unvollständig.

Standardaktion: Die Task wird abnormal beendet.

42 NOSTG

RESP2-Werte:

5

CICS hat keinen Arbeitsspeicher mehr.

Standardaktion: Die Task wird abnormal beendet.

19 NOTOPEN

RESP2-Werte:

6

Die aktuelle CICS-Speicherauszugsdatei (Dataset) ist nicht geöffnet.

Standardaktion: Die Task wird abnormal beendet.

87 OPENERR

RESP2-Werte:

7

Beim Öffnen, Schließen oder Schreiben der aktuellen CICS-Speicherauszugsroutine ist ein Fehler aufgetreten.

Standardaktion: Die Task wird abnormal beendet.

72 SUPPRESSED

RESP2-Werte:

1

Der Transaktionsspeicherauszug wurde durch MAXIMUM in der Tabelle unterdrückt.

2

Der Transaktionsspeicherauszug wurde durch NOTRANDUMP in der Tabelle unterdrückt.

3

Der Transaktionsspeicherauszug wurde durch ein Benutzerexitprogramm unterdrückt.

Standardaktion: Die Task wird abnormal beendet.

Beispiele

Das folgende Beispiel zeigt, wie ein Speicherauszug aller taskbezogenen Speicherbereiche, der Terminalestauertabelle und eines angegebenen Datenbereichs angefordert wird:

```
EXEC CICS DUMP TRANSACTION
          DUMPCODE('name')
          FROM(datenbereich)
          LENGTH(datenwert)
```

Das zweite Beispiel (in PL/I geschrieben) zeigt einen Fall, in dem ein Speicherauszug von fünf taskbezogenen Speicherbereichen erstellt wird:

```
DCL storage_address(5)  POINTER,
    storage_length(5)    FIXED BIN(31),
    nsegs                FIXED BIN(31);
storage_address(1) = ADDR(areal);
storage_length(1)  = CSTG(areal);
:
nsegs = 5;
EXEC CICS DUMP TRANSACTION
          DUMPCODE('name')
          SEGMENTLIST(storage_address)
          LENGTHLIST(storage_length)
          NUMSEGMENTS(nsegs);
```

ENDBR

Beendet das Durchsuchen (Browsing) einer Datei.

ENDBR

➡ ENDBR — FILE(*dateiname*) — REQID(*datenwert*) — SYSID(*systemname*) —

Bedingungen: FILENOTFOUND, ILLOGIC, INVREQ, IOERR, ISCINVREQ, NOTAUTH, SYSIDERR

Dieser Befehl ist threadsicher, wenn für die Datei, auf die er sich bezieht, Folgendes zutrifft:

- Sie ist als fern definiert und der Befehl wird über eine IPIC-Verbindung durch Funktionsverlagerung an eine ferne CICS-Region übertragen.
- Sie ist als lokale VSAM-Datei, als VSAM-RLS-Datei oder als Coupling-Facility-Datentabelle definiert.

Dieser Befehl ist nicht threadsicher, wenn für die Datei, auf die er sich bezieht, Folgendes zutrifft:

- Sie ist als fern definiert und der Befehl wird über eine Nicht-IPIC-Verbindung durch Funktionsverlagerung übertragen.
- Sie ist als gemeinsam genutzte Datentabelle oder als BDAM-Datei definiert.

Beschreibung

Der Befehl **ENDBR** beendet das Durchsuchen (Browsing) für eine Datei oder Datentabelle in einer lokalen oder fernen CICS-Region.

Die Option **UPDATE** ist beim Browsing verfügbar, sodass empfohlen wird, diese zu verwenden, da Sie andernfalls einen Befehl **ENDBR** absetzen müssten, bevor Sie **READ UPDATE** verwenden, um Abbrüche aufgrund von Eigendeadlocks zu vermeiden. Aus ähnlichen Gründen wird empfohlen, einen Befehl **ENDBR** vor Synchronisationspunkten abzusetzen.

Wenn der Befehl **STARTBR** nicht erfolgreich ausgeführt wurde, müssen Sie keinen Befehl **ENDBR** absetzen.

Optionen

FILE(dateiname)

Gibt den Namen einer Datei, die durchsucht wird.

Bei Angabe eines Wertes für **SYSID** wird davon ausgegangen, dass sich die Datei, auf die sich dieser Dateiname bezieht, auf einem fernen System befindet, wobei es ohne Belang ist, ob der Name für CICS definiert ist. Ansonsten wird die Ressourcendefinition für die Datei verwendet, um zu ermitteln, ob sich die Datei (Dataset) auf einem lokalen oder fernen System befindet.

REQID(datenwert)

Gibt eine eindeutige Anforderungs-ID für ein Browsing als Halbwort-Binärwert an, die zur Steuerung mehrerer Browse-Operationen für eine Datei (Dataset) verwendet wird. Falls diese Option nicht angegeben ist, wird vom Standardwert Null ausgegangen.

SYSID(systemname)

Gibt den Namen (1-4 Zeichen) des Systems an, an das die Anforderung übertragen wird.

Bedingungen

12 FILENOTFOUND

RESP2-Werte:

1

Der Name, der in der Option **FILE** angegeben wird, ist in CICS nicht definiert.

Standardaktion: Die Task wird abnormal beendet.

21 ILLOGIC (VSAM)

RESP2-Werte:

110

Es ist ein VSAM-Fehler aufgetreten, der zu keiner der anderen CICS-Antwortkategorien gehört.

Weitere Informationen sind bei **EIBRCODE** im EXEC-Schnittstellenblock verfügbar; Details finden Sie unter [EIB-Felder](#).

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

RESP2-Werte:

35

Der Wert für **REQID**, **SYSID** oder den Dateinamen stimmt nicht mit einem Wert eines erfolgreichen Befehls **STARTBR** überein.

Standardaktion: Die Task wird abnormal beendet.

17 IOERR

RESP2-Werte:

120

Während der Dateisteuerungsoperation ist ein E/A-Fehler aufgetreten. Ein E/A-Fehler ist ein nicht erwartetes Ereignis, das nicht durch eine CICS-Bedingung abgedeckt ist.

Für VSAM-Dateien weist IOERR in der Regel auf einen Hardwarefehler hin. Weitere Informationen sind im EXEC-Schnittstellenblock verfügbar. Einzelheiten finden Sie unter EIB-Felder.

Standardaktion: Die Task wird abnormal beendet.

54 ISCINVREQ

RESP2-Werte:

70

Das ferne System gibt einen Fehler an, der keiner bekannten Bedingung entspricht.

Standardaktion: Die Task wird abnormal beendet.

70 NOTAUTH

RESP2-Werte:

101

Eine Ressourcensicherheitsprüfung ist für FILE(dateiname) fehlgeschlagen.

Standardaktion: Die Task wird abnormal beendet.

53 SYSIDERR

RESP2-Werte:

130

Die Option SYSID gibt einen Namen an, der weder das lokale System noch ein fernes System ist, das durch eine CONNECTION- oder IPCONN-Definition definiert ist. SYSIDERR tritt außerdem auf, wenn die Verbindung zu dem fernen System zwar bekannt, jedoch nicht verfügbar ist. Im Fall einer IPCONN-Verbindung tritt die Bedingung SYSIDERR auf, wenn die Verbindung bekannt ist, jedoch das lokale oder das ferne System die Dateisteuerungsbefehle nicht unterstützt, die durch Funktionsverlagerung über die IP-Interkonnektivität übertragen werden.

Standardaktion: Die Task wird abnormal beendet.

ENDBROWSE ACTIVITY

Beendet das Durchsuchen der untergeordneten Aktivitäten einer BTS-Aktivität oder eines BTS-Prozesses.

ENDBROWSE ACTIVITY

➡ ENDBROWSE — ACTIVITY — BROWSETOKEN(*datenwert*) ➡

Bedingungen: ILLOGIC, TOKENERR

Beschreibung

Der Befehl ENDBROWSE ACTIVITY beendet das Durchsuchen (Browsing) der untergeordneten Aktivitäten einer BTS-Aktivität (oder der untergeordneten Aktivitäten eines BTS-Prozesses) und macht das Browse-Token ungültig.

Optionen

BROWSETOKEN(datenwert)

Gibt das Browse-Token, das gelöscht werden soll, als Vollwort-Binärwert an.

Bedingungen

21 ILLOGIC

RESP2-Werte:

1

Der Wert, der in der Option BROWSETOKEN angegeben wurde, entspricht einem aktuellen Browse-Token, jedoch nicht einem, das zurzeit zum Durchsuchen einer Aktivität verwendet wird.

112 TOKENERR

RESP2-Werte:

3

Das Suchtoken ist ungültig.

ENDBROWSE CONTAINER (BTS)

Beendet das Durchsuchen der Container, die einer BTS-Aktivität zugeordnet sind.

ENDBROWSE CONTAINER

➤ ENDBROWSE — CONTAINER — BROWSETOKEN(*datenwert*) ➤

Bedingungen: ILLOGIC, TOKENERR

Beschreibung

Der Befehl ENDBROWSE CONTAINER beendet das Durchsuchen (Browsing) der Container, die einer BTS-Aktivität oder einem BTS-Prozess zugeordnet sind, und macht das Browse-Token ungültig.

Optionen

BROWSETOKEN(datenwert)

Gibt das Browse-Token, das gelöscht werden soll, als Vollwort-Binärwert an.

Bedingungen

21 ILLOGIC

RESP2-Werte:

1

Der Wert, der in der Option BROWSETOKEN angegeben wurde, entspricht einem aktuellen Browse-Token, jedoch nicht einem, das zurzeit zum Durchsuchen von Containern verwendet wird.

112 TOKENERR

RESP2-Werte:

3

Das Suchtoken ist ungültig.

ENDBROWSE CONTAINER (CHANNEL)

Beendet das Durchsuchen der Container, die einem Kanal zugeordnet sind.

ENDBROWSE CONTAINER

➤ ENDBROWSE — CONTAINER — BROWSETOKEN(*datenwert*) ➤

Bedingungen: ILLOGIC, TOKENERR

Beschreibung

Der Befehl ENDBROWSE CONTAINER (CHANNEL) beendet das Durchsuchen (Browsing) der Container, die einem Kanal zugeordnet sind, und macht das Browse-Token ungültig.

Optionen

BROWSETOKEN(datenwert)

Gibt das Browse-Token, das gelöscht werden soll, als Vollwort-Binärwert an.

Bedingungen

21 ILLOGIC

RESP2-Werte:

1

Der Wert, der in der Option BROWSETOKEN angegeben wurde, entspricht einem aktuellen Browse-Token, jedoch nicht einem, das zurzeit zum Durchsuchen von Containern verwendet wird.

112 TOKENERR

RESP2-Werte:

3

Das Suchtoken ist ungültig.

ENDBROWSE EVENT

Beendet das Durchsuchen der Ereignisse, die einer BTS-Aktivität bekannt sind.

ENDBROWSE EVENT

➤ ENDBROWSE — EVENT — BROWSETOKEN(*datenwert*) ➤

Bedingung: TOKENERR

Beschreibung

Der Befehl ENDBROWSE EVENT beendet das Durchsuchen (Browsing) von Ereignissen, die sich im Gültigkeitsbereich einer BTS-Aktivität befinden, und macht das Browse-Token ungültig.

Optionen

BROWSETOKEN(datenwert)

Gibt das Browse-Token, das gelöscht werden soll, als Vollwort-Binärwert an.

Bedingungen

112 TOKENERR

RESP2-Werte:

3

Das Suchtoken ist ungültig.

ENDBROWSE PROCESS

Beendet das Durchsuchen von Prozessen eines angegebenen Typs im CICS-BTS-System.

ENDBROWSE PROCESS

➤ ENDBROWSE — PROCESS — BROWSETOKEN(*datenwert*) ➤

Bedingungen: ILLOGIC, TOKENERR

Beschreibung

Der Befehl ENDBROWSE PROCESS beendet das Durchsuchen (Browsing) der Prozesse eines angegebenen Typs im CICS-BTS-System und macht das Browse-Token ungültig.

Optionen

BROWSETOKEN(datenwert)

Gibt das Browse-Token, das gelöscht werden soll, als Vollwort-Binärwert an.

Bedingungen

21 ILLOGIC

RESP2-Werte:

1

Der Wert, der in der Option BROWSETOKEN angegeben wurde, entspricht einem aktuellen Browse-Token, jedoch nicht einem, das zurzeit zum Durchsuchen von Prozessen verwendet wird.

112 TOKENERR

RESP2-Werte:

3

Das Suchtoken ist ungültig.

ENDBROWSE TIMER

Beendet einen Suchvorgang für einen BTS-Zeitgeber.

ENDBROWSE TIMER

➤ ENDBROWSE — TIMER — BROWSETOKEN(*datenwert*) ->

Bedingungen: ILLOGIC, INVREQ, TOKENERR

Beschreibung

Der Befehl **ENDBROWSE TIMER** beendet einen Suchvorgang für einen BTS-Zeitgeber.

Optionen

BROWSETOKEN(datenwert)

Gibt als Vollwort-Binärwert das Suchtoken an, das gelöscht werden soll.

Bedingungen

21 ILLOGIC

RESP2-Werte:

1

Der in der Option BROWSETOKEN angegebene Wert stimmt mit einem aktuellen Suchtoken überein, aber nicht mit einem Token, das für einen Suchvorgang eines Zeitgebers verwendet wird.

16 INVREQ

RESP2-Werte:

1

Der Befehl wurde außerhalb des Gültigkeitsbereichs einer momentan aktiven Aktivität abgesetzt.

112 TOKENERR

RESP2-Werte:

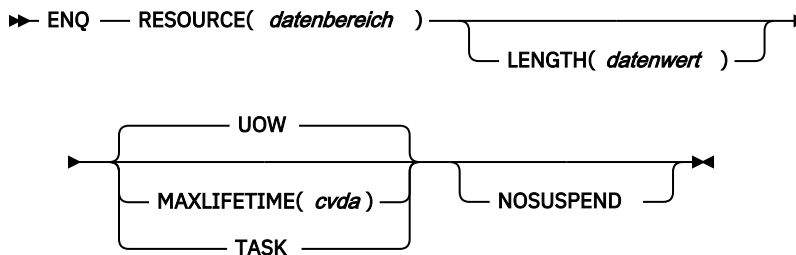
3

Das Suchtoken ist ungültig.

ENQ

Terminiert die Verwendung einer Ressource durch eine Task (Einreihen in die Warteschlange).

ENQ



Bedingungen: ENQBUSY, INVREQ, LENGERR

Dieser Befehl ist threadsicher, wenn er als lokal definiert wird. Er ist nicht threadsicher, wenn er als global definiert wird.

Beschreibung

Der Befehl ENQ sorgt dafür, dass die weitere Ausführung der Task, die den Befehl ENQ absetzt, mit der Verfügbarkeit der angegebenen Ressource synchronisiert wird. Die Steuerung wird an die Task zurückgegeben, wenn die Ressource verfügbar ist.

Eine Ressource im Kontext dieses Befehls ist jede 1 bis 255 Byte lange Zeichenfolge, die durch unternehmensinterne Standards definiert wurde, um gegen widersprüchliche Aktionen zwischen Tasks zu schützen oder um nur einen Thread innerhalb eines Programms zu verwenden.

Wenn sich eine Task mit dem Befehl ENQ in die Warteschlange für eine Ressource einreicht, sich jedoch nicht aus der Warteschlange entfernt, gibt CICS die Ressource automatisch bei der Synchronisationspunktverarbeitung (einschließlich DL/I-, PCB- und TERM-Aufrufen) oder bei Beendigung der Task frei. Die Option UOW erzwingt das Entfernen aus der Warteschlange am Ende einer Arbeitseinheit (UOW). Die Option TASK erzwingt das Entfernen aus der Warteschlange am Ende einer Task. Wenn mehrere Arbeitseinheiten in einer Task enthalten sind, werden die Arbeitseinheiten durch die Einreihung der Task nacheinander übernommen.

Wenn mehr als ein Befehl ENQ für dieselbe Ressource durch eine bestimmte Task abgesetzt wurde, bleibt die Ressource so lange Eigentum dieser Task, bis die Task eine entsprechende Anzahl von Befehlen DEQ abgesetzt hat.

Die Ressource, für die sich die Task in die Warteschlange einreicht, muss durch eine der folgenden Methoden angegeben werden:

- Angeben eines Datenbereichs, der die Ressource ist. Die Position (Adresse) des Datenbereichs im Speicher ist relevant, nicht sein Inhalt.
- Angeben eines Datenbereichs, der ein eindeutiges Argument enthält (z. B. einen Mitarbeiternamen), das die Ressource darstellt. Der Inhalt des Datenwerts ist relevant, nicht seine Position. Die Option LENGTH ist erforderlich. Das Vorhandensein der Option LENGTH weist CICS an, sich für den Inhalt des Datenwerts in die Warteschlange einzureihen.

Wenn ein Befehl EXEC CICS ENQ (oder DEQ) für eine Ressource abgesetzt wird, deren Name dem einer installierten ENQMODEL-Ressourcendefinition entspricht, prüft CICS den Wert des Attributs ENQSCOPE, um zu ermitteln, ob der Bereich lokal oder der gesamte Sysplex ist. Wenn das Attribut ENQSCOPE nicht leer ist, verarbeitet CICS die Befehle ENQ oder DEQ für den gesamten Sysplex und übergibt den Warteschlangennamen und den Ressourcennamen an z/OS Global Resource Serialization, um die Einreihung zu verwalten. Falls das Attribut ENQSCOPE leer gelassen wurde (Standardwert), verarbeitet CICS den Befehl ENQ als lokal für die absetzende CICS-Region. Wenn keine ENQMODEL-Definition dem Ressourcennamen entspricht, ist der Bereich des Befehls ENQ lokal. Weitere Informationen zu ENQMODEL-Ressourcen finden Sie unter [ENQMODEL-Ressourcen](#).

Nichtverfügbarkeit von Ressourcen

Wenn eine Ressource nicht verfügbar ist, wenn der Befehl ENQ abgesetzt wird, wird das Anwendungsprogramm ausgesetzt, bis sie verfügbar wird. Wenn jedoch die Option NOSUSPEND angegeben wurde und

die Ressource nicht verfügbar ist, wird die Bedingung ENQBUSY ebenso ausgelöst, wie wenn Sie eine aktive HANDLE-Bedingung haben. Dies gibt dem Anwendungsprogramm die Möglichkeit, den Fall einer nicht verfügbaren Ressource (durch einen Befehl HANDLE CONDITION ENQBUSY) zu verarbeiten, ohne auf die Verfügbarkeit der Ressource zu warten.

Optionen

LENGTH(*datenwert*)

Gibt einen Halbwort-Binärwert für die Länge der Ressource an, für die sich die Task in die Warteschlange einreihen soll. Der Wert muss im Bereich von 1 bis 255 liegen. Andernfalls tritt die Bedingung LENGERR auf. Wenn die Option LENGTH in einem Befehl ENQ angegeben wird, muss sie auch im Befehl DEQ für diese Ressource angegeben werden und die Werte dieser Optionen müssen übereinstimmen. Sie müssen die Option LENGTH angeben, wenn Sie die Methode verwenden, bei der ein Datenwert mit einem eindeutigen Argument angegeben wird. Für die Methode, bei der ein Datenbereich als Ressource angegeben wird, müssen Sie LENGTH nicht angeben. An der vorhandenen bzw. nicht vorhandenen Angabe von LENGTH erkennt CICS, welche Methode Sie verwenden.

MAXLIFETIME(*cvda*)

Gibt die Dauer der Einreihung (ENQ) für die automatische Freigabe durch CICS an. Die CVDA-Werte lauten wie folgt:

UOW

Die Dauer der Einreihung (ENQ) ist eine Arbeitseinheit (UOW). Beispiele hierfür sind ein Befehl SYNCPOINT ROLLBACK oder SYNCPOINT, wenn die Anwendung keinen Befehl DEQ vor dem Ende der Arbeitseinheit absetzt. Dies ist der Standardwert.

Anmerkung: Aus Kompatibilitätsgründen wird auch der CVDA-Wert LUW unterstützt.

TASK

Die Dauer der Einreihung (ENQ) ist eine Task. Die Einreihung übernimmt die Arbeitseinheiten in der Task. Verwenden Sie die Option MAXLIFETIME(TASK) mit Sorgfalt, weil andere Tasks, die Befehle ENQ für dieselbe Ressource absetzen, bis zum Ende dieser Task ausgesetzt werden könnten.

Es gibt zwei Möglichkeiten, diese Option zu codieren.

- Sie können einen CVDA-Wert mit der Umsetzerroutine DFHVALUE zuweisen. Dadurch können Sie einen CVDA-Wert im Programm ändern. Beispiel:

```
MOVE DFHVALUE(UOW) TO AREA-A
EXEC CICS ENQ RESOURCE(RESNAME)
        MAXLIFETIME(AREA-A)
```

- Wenn die erforderliche Aktion immer die gleiche ist, können Sie den Wert direkt deklarieren. Beispiel:

```
EXEC CICS ENQ RESOURCE(RESNAME) UOW
```

oder

```
EXEC CICS ENQ RESOURCE(RESNAME) TASK
```

NOSUSPEND

Gibt an, dass das Anwendungsprogramm nicht ausgesetzt wird, wenn die Ressource, die im Befehl ENQ angegeben wird, nicht verfügbar ist, sondern die Bedingung ENQBUSY auftritt.

Beachten Sie jedoch, dass bei einem aktiven Befehl HANDLE CONDITION für die Bedingung ENQBUSY, wenn der Befehl ausgeführt wird, die Steuerung an die Benutzerbezeichnung übergeben wird, die im Befehl HANDLE CONDITION angegeben wurde. Dieses Verhalten hat Vorrang vor der Option NOSUSPEND, wird jedoch natürlich durch NOHANDLE oder RESP zurückgewiesen.

RESOURCE(*datenbereich*)

Gibt die Ressource, für die die Task in die Warteschlange gestellt wird, wie folgt an:

- Angabe eines Bereichs, dessen Adresse die Ressource darstellt.

- Angabe einer Variablen, die die Ressource (z. B. einen Mitarbeiternamen) enthält. In diesem Fall müssen Sie die Option LENGTH verwenden.

Bedingungen

55 ENQBUSY

Tritt auf, wenn ein Befehl ENQ eine Ressource angibt, die nicht verfügbar ist, und die Option NOSUSPEND angegeben wurde oder wenn ein aktiver Befehl HANDLE CONDITION ENQBUSY vorhanden ist.

Wenn die Option NOSUSPEND nicht angegeben wird und der Befehl ENQ eine Ressource angibt, die nicht verfügbar ist, wird das Anwendungsprogramm ausgesetzt und die Bedingung ENQBUSY wird nicht ausgelöst.

Standardaktion: Die Bedingung wird ignoriert.

16 INVREQ

Die RESP2-Werte lauten wie folgt:

2

Die Option MAXLIFETIME ist mit einem falschen CVDA-Wert festgelegt.

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

RESP2-Werte:

1

Der Wert, der für die Option LENGTH angegeben wurde, liegt außerhalb des Bereichs von 1 bis 255.

Standardaktion: Die Task wird abnormal beendet.

Beispiele

Zwei Tasks, die sich für dieselbe Ressource in die Warteschlange einreihen und einen Datenbereich angeben, der die Ressource ist, müssen auf dieselbe Position im Speicher verweisen. Sie könnten beispielsweise auf dieselbe Position im gemeinsamen Arbeitsbereich (CWA) verweisen.

```
EXEC CICS ENQ
      RESOURCE(RESNAME)
```

Zwei Tasks, die sich für dieselbe Ressource in die Warteschlange einreihen und einen Datenbereich angeben, der ein eindeutiges Argument enthält, können auf dieselbe Position oder auf verschiedene Positionen im Speicher, jedoch muss der Inhalt der Positionen übereinstimmen. Die Länge muss in der Option LENGTH angegeben werden.

```
EXEC CICS ENQ
      RESOURCE(SOCSECNO)
      LENGTH(9)
```

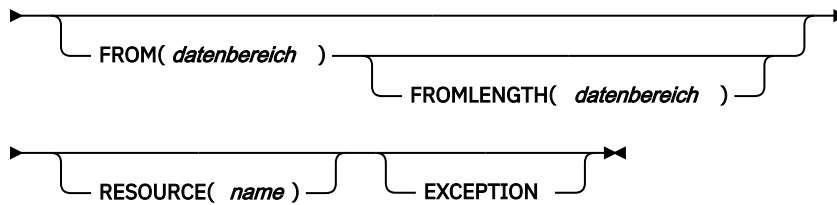
Die beiden Methoden können nicht kombiniert werden. Wenn die eine Task die Option LENGTH verwendet, die andere Task jedoch nicht, betrachtet CICS die Einreihung mit der Option LENGTH und die Einreihung ohne die Option LENGTH als verschiedene Typen von Einreihungen und die Tasks werden nicht serialisiert.

ENTER TRACENUM

Schreibt einen Traceeintrag.

ENTER TRACENUM

➔ ENTER TRACENUM(*datenwert*) ➔



Bedingungen: INVREQ, LENGERR

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl ENTER TRACENUM erstellt einen Traceeintrag in den momentan aktiven Tracezielen. CICS schreibt den Traceeintrag nur, wenn die Trace-Flags für Master und Benutzer aktiviert (ON) sind. Bei Angabe der Option EXCEPTION wird der Traceeintrag allerdings immer geschrieben, auch wenn die Trace-Flags für Master und Benutzer inaktiv (OFF) sind. Einträge für den Ausnahmebedingungs-Trace werden immer in die interne Tracetabelle geschrieben (auch wenn das interne Tracing nicht aktiviert ist), jedoch werden sie in andere Ziele geschrieben, und dies nur, wenn diese aktiv sind.

Sie können die Option für Ausnahmebedingungs-Trace (EXCEPTION) in einem Anwendungsprogramm verwenden, um einen Traceeintrag zu schreiben, wenn das Programm eine Ausnahmebedingung oder eine abnormale Bedingung erkennt. Zu diesem Zweck schließen Sie einen Befehl ENTER TRACENUM(datenwert) EXCEPTION in die Routine für die Behandlung von abnormalen Bedingungen und Fehlern Ihres Programms ein.

Wenn Sie einen Eintrag für den Ausnahmebedingungs-Trace in einer Fehlersituation schreiben wollen, in der ein Anwendungsprogramm die Steuerung abgegeben hat, können Sie einen Befehl ENTER TRACENUM(datenwert) EXCEPTION in einem benutzergeschriebenen Programmfehlerprogramm (PEP - Program Error Program) absetzen. Programmierinformationen zum Ändern des Programms DFHPEP finden Sie unter [Programmfehlerprogramm schreiben](#).

Informationen zum Traceeintragsformat finden Sie unter [CICS-Trace verwenden](#).

Optionen

EXCEPTION

Gibt an, dass CICS einen Benutzerausnahmebedingungs-Traceeintrag schreiben soll. Die Option EXCEPTION überschreibt das Masterbenutzer-Trace-Flag und CICS schreibt den Traceeintrag, auch wenn das Benutzer-Trace-Flag inaktiviert ist. Einträge in Ausnahmebedingungs-Traces werden durch die Zeichen *EXCU gekennzeichnet, wenn die Traceeinträge durch das Tracedienstprogramm formatiert werden. Weitere Informationen zu Benutzerausnahmebedingungs-Traceeinträgen finden Sie unter [CICS-Ausnahmetracing](#).

FROM(datenbereich)

Gibt einen Datenbereich an, dessen Inhalt in das Datenfeld des Tracetabloneintrags eingegeben werden soll. Wenn Sie die Option FROM nicht angeben, werden zwei Vollwörter an binären Nullen übergeben.

FROMLENGTH(datenbereich)

Gibt einen Datenbereich, der die Länge der Tracedaten enthält, mit einem Halbwort-Binärwert aus dem Bereich von 0 - 4000 Byte an. Wenn FROMLENGTH nicht angegeben wird, wird eine Länge von 8 Byte angenommen.

RESOURCE(name)

Gibt einen acht Zeichen langen Namen an, der in das Ressourcenfeld des Tracetabloneintrags eingegeben werden soll. Verwenden Sie einen Namen, der den Benutzer-Trace identifiziert, zum Beispiel den Namen des aufrufenden Programms.

TRACENUM(datenwert)

Gibt die Trace-ID für den Benutzertracetableneintrag als Halbwort-Binärwert aus dem Bereich von 0 bis 199 an. Trace-IDs können zum Filtern eines Speicherauszugs verwendet werden, sodass es sinnvoll ist, einen Wert anzugeben, der ein geeignetes Filtern ermöglicht. Weitere Informationen finden Sie unter Speicherauszüge bei der Fehlerbestimmung verwenden.

Es ist empfehlenswert, eine Nummer anzugeben, die nicht bereits im Gebrauch ist. Die Nummern von 1 bis 20 sind beispielsweise im Gebrauch, wenn CICSplex SM installiert ist. Die folgenden Nummern können im Gebrauch sein, wenn verschiedene CICS-Beispielprogramme installiert sind: 4-8, 11-13, 15-20, 27, 33, 35-37, 50-52, 77, 101-102, 190. Prüfen Sie außerdem Ihr eigenes System auf Benutzertracenummern, die bereits im Gebrauch sind.

Bedingungen

16 INVREQ

RESP2-Werte:

1

Der Wert für TRACENUM liegt außerhalb des Bereichs von 0 bis 199.

2

Es ist kein gültiges Traceziel vorhanden.

3

Das Benutzertrace-Flag ist auf OFF gesetzt und die Option EXCEPTION wurde nicht angegeben.

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

RESP2-Werte:

4

Der Wert für FROMLENGTH liegt außerhalb des Bereichs von 0 bis 4000.

Standardaktion: Die Task wird abnormal beendet.

Beispiele

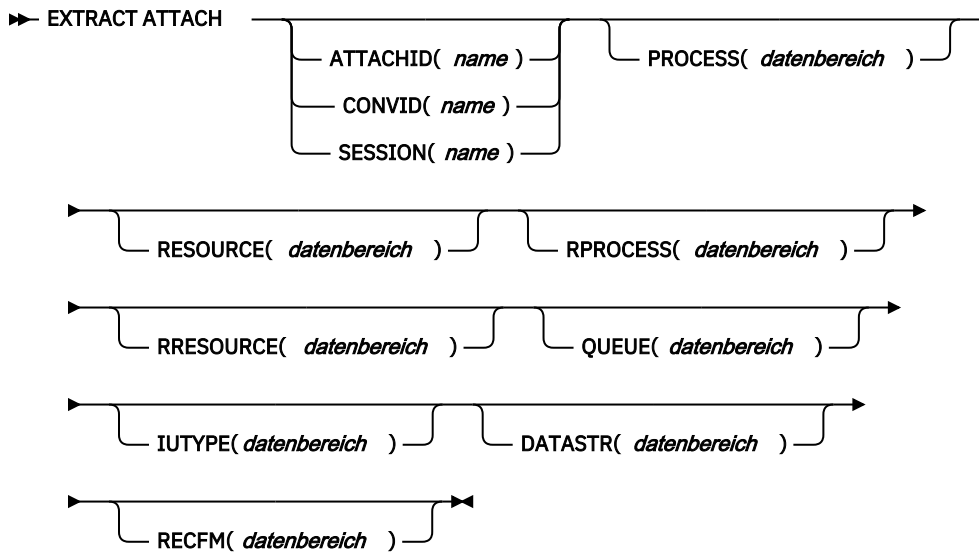
Das folgende COBOL-Beispiel zeigt, wie ein Benutzertraceeintrag mit der Trace-ID 123 mit Tracedaten aus einem Datenbereich mit dem Namen USER-TRACE-ENTRY geschrieben wird:

```
EXEC CICS ENTER TRACENUM(123)
      FROM(USER-TRACE-ENTRY)
END-EXEC.
```

EXTRACT ATTACH (LUTYPE6.1)

Ruft Werte aus einem Anschlussheader vom LU-Typ 6.1 (LUTYPE6.1) ab.

EXTRACT ATTACH (LUTYPE6.1)



Bedingungen: CBIDERR, INVREQ, NOTALLOC

Beschreibung

Der Befehl EXTRACT ATTACH ruft eine Gruppe von Werten ab, die in einem Anschlussheadersteuerblock (Attach Header Control Block) gespeichert sind oder die zuvor erstellt wurden. Damit der Befehl Informationen aus einem empfangenen Funktionsverwaltungsheader (FMH - Function Management Header) für den Anschluss (ATTACH) abrufen kann, muss EIBATT mit einem Befehl RECEIVE oder CONVERSE festgelegt worden sein.

Optionen

ATTACHID(name)

Gibt an, dass Werte aus einem Anschlussheadersteuerblock abgerufen werden sollen. Der Name (1 - 8 Zeichen) gibt diesen Steuerblock für die lokale Task an.

CONVID(name)

Gibt den Dialog an, auf den sich der Befehl bezieht. Der vier Zeichen lange Name bezeichnet entweder das Token, das von einem zuvor ausgeführten Befehl ALLOCATE in EIBRSRCE im EIB zurückgegeben wurde, oder das Token, das die Hauptsitzung (die durch einen zuvor ausgeführten Befehl ASSIGN zurückgegeben wurde) darstellt.

DATASTR(datenbereich)

Entspricht dem Datenstromprofilfeld ATTDSP in einem Anschlussfunktionsverwaltungsheader für LU-TYPE6.1.

Für die Kommunikation zwischen zwei CICS-Systemen hat das Datenstromprofilfeld in einem Anschlussfunktionsverwaltungsheader für CICS keine besondere Bedeutung. Für die meisten CICS-Anwendungen kann die Option weggelassen werden.

Der im Datenbereich zurückgegebene Wert ist ein Halbwort-Binärwert. Nur das niedrigstwertige Byte wird verwendet. Die SNA-definierten Bedeutungen der Bits sind folgende:

0-7	Reserviert - müssen auf null gesetzt werden.
8-11	0000 - Benutzerdefiniert. 1111 - SCS-Datenstrom. 1110 - 3270-Datenstrom. 1101 - Strukturiertes Feld. 1100 - Verwaltung logischer Datensätze.
12-15	Durch Benutzer definiert, wenn Bits 8-11 auf 0000 gesetzt sind; ansonsten reserviert. (Müssen auf null gesetzt werden.)

IUTYPE(datenbereich)

Entspricht dem Austauschheitenfeld ATTU in einem Anschlussfunktionsverwaltungsheader für LUTYPE6.1.

Für die Kommunikation zwischen zwei CICS-Systemen hat das Austauschheitenfeld in einem Anschlussfunktionsverwaltungsheader für CICS keine besondere Bedeutung. Für die meisten CICS-Anwendungen kann die Option weggelassen werden.

Der im Datenbereich zurückgegebene Wert ist ein Halbwort-Binärwert. Nur die 7 niedrigstwertigen Byte werden verwendet. Die SNA-definierten Bedeutungen der Bits sind folgende:

0-10	Reserviert - müssen auf null gesetzt werden.
11	0 - Nicht Ende der Mehrfachketten-Austauscheinheit. 1 - Ende der Mehrfachketten-Austauscheinheit.
12,13	Reserviert - müssen auf null gesetzt werden.
14,15	00 - Mehrfachketten-Austauscheinheit. 01 - Einzelketten-Austauscheinheit. 10 - Reserviert. 11 - Reserviert.

PROCESS(datenbereich)

Entspricht dem Prozessnamen ATTDPN in einem Anschlussfunktionsverwaltungsheader für LUTYPE6.1.

Für die Kommunikation zwischen zwei CICS-Systemen kann eine Transaktion, die in dem einen System ausgeführt wird, eine Sitzung zu dem zweiten System anfordern und die Transaktion angeben, die angeschlossen (Attach) werden soll. In dem zweiten System wird die Identifikation in der ersten Kette von Daten transportiert, die über die Sitzung gesendet wird.

Im Allgemeinen identifizieren die ersten vier Byte von Daten die anzuschließende Transaktion. Jedoch kann ein Anschlussfunktionsverwaltungsheader (FHM), der die anzuschließende Transaktion identifiziert, erstellt und gesendet werden. Das empfangende CICS-System verwendet nur die ersten vier Byte des Prozessnamens als Transaktionsnamen.

Von CICS wird Prozessnamen in Anschlussfunktionsverwaltungsheadern, die in anderen Ketten als der ersten Kette von Daten gesendet werden, keine Bedeutung beigemessen.

Informationen zur Kommunikation zwischen einem CICS-System und einem anderen Subsystem finden Sie in der für das Subsystem bereitgestellten Dokumentation zur Verwendung des Prozessnamensfelds in einem Anschlussfunktionsverwaltungsheader (FMH).

QUEUE(datenbereich)

Entspricht dem Warteschlangennamen ATTDQN in einem Anschlussfunktionsverwaltungsheader.

Für die Kommunikation zwischen zwei CICS-Systemen hat der Warteschlangenname in einem Anschlussfunktionsverwaltungsheader für CICS keine Bedeutung.

Informationen zur Kommunikation zwischen einem CICS-System und einem anderen Subsystem finden Sie in der für das Subsystem bereitgestellten Dokumentation zur Verwendung des Warteschlangennamensfelds in einem Anschlussfunktionsverwaltungsheader (FMH).

RECFM(datenbereich)

Entspricht dem Feld für den Entblockungsalgorithmus ATTDDBA in einem Anschlussfunktionsverwaltungsheader für LUTYPE6.1.

Für die Kommunikation zwischen zwei CICS-Systemen hat das Feld für den Entblockungsalgorithmus in einem Anschlussfunktionsverwaltungsheader für CICS keine besondere Bedeutung. Für die meisten CICS-Anwendungen kann die Option weggelassen werden.

Informationen zur Kommunikation zwischen einem CICS-System und einem anderen Subsystem finden Sie in der für das Subsystem bereitgestellten Dokumentation zur Verwendung des Austausch-einheitenfelds in einem Anschlussfunktionsverwaltungsheder (FMH).

Der im Datenbereich zurückgegebene Wert ist ein Halbwort-Binärwert. Nur das niedrigstwertige Byte wird verwendet. Die SNA-definierten Bedeutungen der Bits sind folgende:

```
0-7   Reserviert - müssen auf null gesetzt werden.  
8-15  X'00' - Reserviert.  
      X'01' - Variable Länge.  
           Variabel geblockt.  
      X'02' - Reserviert.  
      X'03' - Reserviert.  
      X'04' - Kette von RUs (Anforderung/Antwort-Einheiten).  
      X'05' bis X'FF' - Reserviert.
```

RESOURCE(datenbereich)

Entspricht dem Ressourcennamen ATTPRN in einem Anschlussfunktionsverwaltungsheder für LU-TYPE6.1.

Für die Kommunikation zwischen zwei CICS-Systemen hat der Ressourcename in einem Anschlussfunktionsverwaltungsheder für CICS keine Bedeutung.

Informationen zur Kommunikation zwischen einem CICS-System und einem anderen Subsystem finden Sie in der für das Subsystem bereitgestellten Dokumentation zur Verwendung des Felds für den Ressourcennamen in einem Anschlussfunktionsverwaltungsheder (FMH).

RPROCESS(datenbereich)

Entspricht dem Rückgabeprozessnamen ATTRDPN in einem Anschlussfunktionsverwaltungsheder für LUTYPE6.1.

Für die Kommunikation zwischen zwei CICS-Systemen hat der Rückgabeprozessname in einem Anschlussfunktionsverwaltungsheder für CICS keine Bedeutung.

Informationen zur Kommunikation zwischen einem CICS-System und einem anderen Subsystem finden Sie in der für das Subsystem bereitgestellten Dokumentation zur Verwendung des Felds für den Rückgabeprozessnamen in einem Anschlussfunktionsverwaltungsheder (FMH).

RRESOURCE(datenbereich)

Entspricht dem Rückgaberessourcennamen ATTPRN in einem Anschlussfunktionsverwaltungsheder für LUTYPE6.1.

Für die Kommunikation zwischen zwei CICS-Systemen hat der Rückgaberessourcename in einem Anschlussfunktionsverwaltungsheder für CICS keine Bedeutung.

Informationen zur Kommunikation zwischen einem CICS-System und einem anderen Subsystem finden Sie in der für das Subsystem bereitgestellten Dokumentation zur Verwendung des Felds für den Rückgaberessourcennamen in einem Anschlussfunktionsverwaltungsheder (FMH).

SESSION(name)

Gibt die symbolische ID (1 - 4 Zeichen) für einen Terminaleintrag in der Terminalsteuertabelle (TCTTE) für die Sitzung an. Diese Option gibt die alternative Funktion an, die verwendet werden soll.

Bedingungen

62 CBIDERR

Tritt auf, wenn der angeforderte Anschlussheadersteuerblock nicht gefunden werden kann.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

Tritt auf, wenn falsche Daten gefunden werden.

Standardaktion: Die Task wird abnormal beendet.

61 NOTALLOC

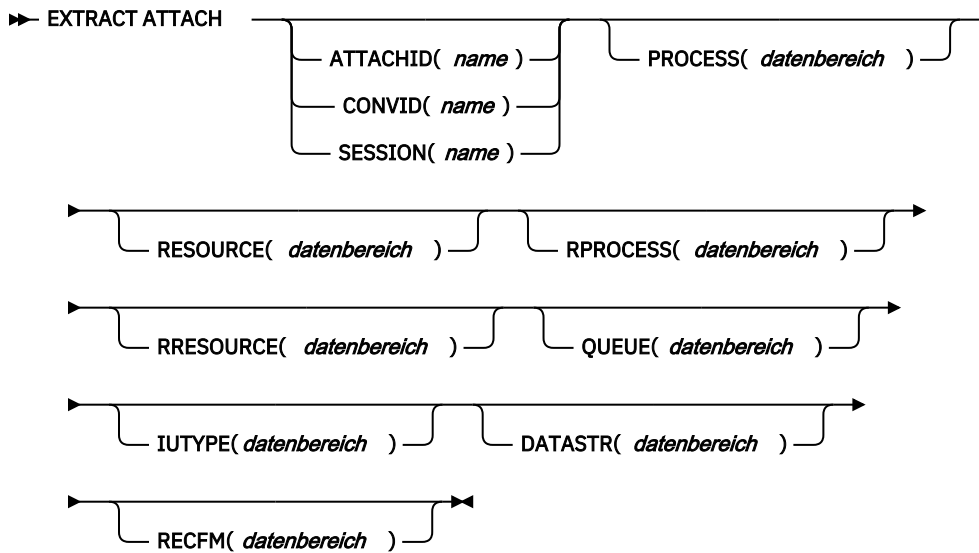
Tritt auf, wenn die im Befehl angegebene Funktion (Facility) nicht der Anwendung gehört.

Standardaktion: Die Task wird abnormal beendet.

EXTRACT ATTACH (MRO)

Ruft Werte aus einem MRO-Anschlussheader ab.

EXTRACT ATTACH (MRO)



Bedingungen: CBIDERR, INVREQ, NOTALLOC

Beschreibung

Der Befehl EXTRACT ATTACH ruft eine Gruppe von Werten ab, die in einem Anschlussheadersteuerblock (Attach Header Control Block) gespeichert sind oder die zuvor erstellt wurden. Damit der Befehl Informationen aus einem empfangenen Funktionsverwaltungsheader (FMH - Function Management Header) für den Anschluss (ATTACH) abrufen kann, muss EIBATT mit einem Befehl RECEIVE oder CONVERSE festgelegt worden sein.

Weitere Informationen zu MRO und IRC finden Sie unter [Einführung in die übergreifende Kommunikation in CICS](#).

Optionen

ATTACHID(name)

Gibt an, dass Werte aus einem Anschlussheadersteuerblock abgerufen werden sollen. Der Name (1-8 Zeichen) gibt diesen Steuerblock für die lokale Task an.

CONVID(name)

Gibt den Dialog an, auf den sich der Befehl bezieht. Der vier Zeichen lange Name bezeichnet entweder das Token, das von einem zuvor ausgeführten Befehl ALLOCATE in EIBRSRCE im EIB zurückgegeben wurde, oder das Token, das die Hauptsitzung (die durch einen zuvor ausgeführten Befehl ASSIGN zurückgegeben wurde) darstellt.

DATASTR(datenbereich)

Entspricht dem Datenstromprofilfeld ATTDSP in einem Anschlussfunktionsverwaltungsheader für LU-TYPE6.1.

Für die Kommunikation zwischen zwei CICS-Systemen hat das Datenstromprofilfeld in einem Anschlussfunktionsverwaltungsheader für CICS keine besondere Bedeutung. Für die meisten CICS-Anwendungen kann die Option weggelassen werden.

Der im Datenbereich zurückgegebene Wert ist ein Halbwort-Binärwert. Nur das niedrigstwertige Byte wird verwendet. Die SNA-definierten Bedeutungen der Bits sind folgende:

0-7	Reserviert - müssen auf null gesetzt werden.
8-11	0000 - Benutzerdefiniert.
	1111 - SCS-Datenstrom.
	1110 - 3270-Datenstrom.
	1101 - Strukturiertes Feld.
	1100 - Verwaltung logischer Datensätze.
12-15	Durch Benutzer definiert, wenn Bits 8-11 auf 0000 gesetzt sind; ansonsten reserviert. (Müssen auf null gesetzt werden.)

IUTYPE(datenbereich)

Entspricht dem Austauschheitenfeld ATTU in einem Anschlussfunktionsverwaltungsheader für LUTYPE6.1.

Für die Kommunikation zwischen zwei CICS-Systemen hat das Austauschheitenfeld in einem Anschlussfunktionsverwaltungsheader für CICS keine besondere Bedeutung. Für die meisten CICS-Anwendungen kann die Option weggelassen werden. Der im Datenbereich zurückgegebene Wert ist ein Halbwort-Binärwert. Nur die 7 niedrigstwertigen Byte werden verwendet. Die SNA-definierten Bedeutungen der Bits sind folgende:

0-10	Reserviert - müssen auf null gesetzt werden.
11	0 - Nicht Ende der Mehrfachketten-Austauscheinheit.
	1 - Ende der Mehrfachketten-Austauscheinheit.
12,13	Reserviert - müssen auf null gesetzt werden.
14,15	00 - Mehrfachketten-Austauscheinheit.
	01 - Einzelketten-Austauscheinheit.
	10 - Reserviert.
	11 - Reserviert.

PROCESS(datenbereich)

Entspricht dem Prozessnamen ATTDPN in einem Anschlussfunktionsverwaltungsheader für LUTYPE6.1.

Für die Kommunikation zwischen zwei CICS-Systemen kann eine Transaktion, die in dem einen System ausgeführt wird, eine Sitzung zu dem zweiten System anfordern und die Transaktion angeben, die angeschlossen (Attach) werden soll. In dem zweiten System wird die Identifikation in der ersten Kette von Daten transportiert, die über die Sitzung gesendet wird.

Im Allgemeinen identifizieren die ersten vier Byte von Daten die anzuschließende Transaktion. Jedoch kann ein Anschlussfunktionsverwaltungsheader (FHM), der die anzuschließende Transaktion identifiziert, erstellt und gesendet werden. Das empfangende CICS-System verwendet nur die ersten vier Byte des Prozessnamens als Transaktionsnamen. Von CICS wird Prozessnamen in Anschlussfunktionsverwaltungsheadern, die in anderen Ketten als der ersten Kette von Daten gesendet werden, keine Bedeutung beigemessen.

Informationen zur Kommunikation zwischen einem CICS-System und einem anderen Subsystem finden Sie in der für das Subsystem bereitgestellten Dokumentation zur Verwendung des Prozessnamensfelds in einem Anschlussfunktionsverwaltungsheader (FMH).

QUEUE(datenbereich)

Entspricht dem Warteschlangennamen ATTDQN in einem Anschlussfunktionsverwaltungsheader. Für die Kommunikation zwischen zwei CICS-Systemen hat der Warteschlangennamen in einem Anschlussfunktionsverwaltungsheader für CICS keine Bedeutung.

Informationen zur Kommunikation zwischen einem CICS-System und einem anderen Subsystem finden Sie in der für das Subsystem bereitgestellten Dokumentation zur Verwendung des Warteschlangennamensfelds in einem Anschlussfunktionsverwaltungsheader (FMH).

RECFM(datenbereich)

Entspricht dem Feld für den Entblockungsalgorithmus ATTDDBA in einem Anschlussfunktionsverwaltungsheader für LUTYPE6.1.

Für die Kommunikation zwischen zwei CICS-Systemen hat das Feld für den Entblockungsalgorithmus in einem Anschlussfunktionsverwaltungsheader für CICS keine besondere Bedeutung. Für die meisten CICS-Anwendungen kann die Option weggelassen werden.

Informationen zur Kommunikation zwischen einem CICS-System und einem anderen Subsystem finden Sie in der für das Subsystem bereitgestellten Dokumentation zur Verwendung des Austausch-einheitenfelds in einem Anschlussfunktionsverwaltungsheder (FMH).

Der im Datenbereich zurückgegebene Wert ist ein Halbwort-Binärwert. Nur das niedrigstwertige Byte wird verwendet. Die SNA-definierten Bedeutungen der Bits sind folgende:

```
0-7   Reserviert - müssen auf null gesetzt werden.
8-15  X'00' - Reserviert.
      X'01' - Variable Länge.
           Variabel geblockt.
      X'02' - Reserviert.
      X'03' - Reserviert.
      X'04' - Kette von RUs (Anforderung/Antwort-Einheiten).
      X'05' bis X'FF' - Reserviert.
```

RESOURCE(datenbereich)

Entspricht dem Ressourcennamen ATTPRN in einem Anschlussfunktionsverwaltungsheder für LU-TYPE6.1.

Für die Kommunikation zwischen zwei CICS-Systemen hat der Ressourcename in einem Anschlussfunktionsverwaltungsheder für CICS keine Bedeutung.

Informationen zur Kommunikation zwischen einem CICS-System und einem anderen Subsystem finden Sie in der für das Subsystem bereitgestellten Dokumentation zur Verwendung des Felds für den Ressourcennamen in einem Anschlussfunktionsverwaltungsheder (FMH).

RPROCESS(datenbereich)

Entspricht dem Rückgabeprozessnamen ATTRDPN in einem Anschlussfunktionsverwaltungsheder für LUTYPE6.1.

Für die Kommunikation zwischen zwei CICS-Systemen hat der Rückgabeprozessname in einem Anschlussfunktionsverwaltungsheder für CICS keine Bedeutung.

Informationen zur Kommunikation zwischen einem CICS-System und einem anderen Subsystem finden Sie in der für das Subsystem bereitgestellten Dokumentation zur Verwendung des Felds für den Rückgabeprozessnamen in einem Anschlussfunktionsverwaltungsheder (FMH).

RRESOURCE(datenbereich)

Entspricht dem Rückgaberessourcennamen ATTRPRN in einem Anschlussfunktionsverwaltungsheder für LUTYPE6.1.

Für die Kommunikation zwischen zwei CICS-Systemen hat der Rückgaberessourcename in einem Anschlussfunktionsverwaltungsheder für CICS keine Bedeutung.

Informationen zur Kommunikation zwischen einem CICS-System und einem anderen Subsystem finden Sie in der für das Subsystem bereitgestellten Dokumentation zur Verwendung des Felds für den Rückgaberessourcennamen in einem Anschlussfunktionsverwaltungsheder (FMH).

SESSION(name)

Gibt die symbolische ID (1-4 Zeichen) für einen Terminaleintrag in der Terminalsteuertabelle (TCTTE) für die Sitzung an. Diese Option gibt die zu verwendende alternative Funktion an.

Bedingungen

62 CBIDERR

Tritt auf, wenn der angeforderte Anschlussheadersteuerblock nicht gefunden werden kann.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

Tritt auf, wenn falsche Daten gefunden werden.

Standardaktion: Die Task wird abnormal beendet.

61 NOTALLOC

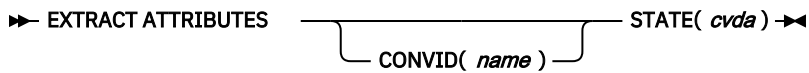
Tritt auf, falls die Anwendung nicht Eigner der im Befehl angegebenen Funktion ist.

Standardaktion: Die Task wird abnormal beendet.

EXTRACT ATTRIBUTES (APPC)

Ruft den Status des APPC-Dialogs ab.

EXTRACT ATTRIBUTES (APPC)



Bedingungen: INVREQ, NOTALLOC

Beschreibung

Der Befehl EXTRACT ATTRIBUTES extrahiert Dialogstatusinformationen für formatfreie APPC-Dialoge.

Optionen

CONVID(name)

Gibt den Dialog an, auf den sich der Befehl bezieht. Der vier Zeichen lange Name bezeichnet entweder das Token, das von einem zuvor ausgeführten Befehl ALLOCATE in EIBRSRCE im EIB zurückgegeben wurde, oder das Token, das die Hauptsitzung (die durch einen zuvor ausgeführten Befehl ASSIGN zurückgegeben wurde) darstellt.

Aus Gründen der Kompatibilität mit früheren Releases wird SESSION als Synonym für CONVID akzeptiert. Neue Programme sollten CONVID verwenden.

Standardmäßig wird die Hauptfunktion angenommen.

STATE(cvda)

Ruft den Status des Transaktionsprogramms ab. Die folgenden CVDA-Werte können von CICS zurückgegeben werden:

- ALLOCATED
- CONFFREE
- CONFRECEIVE
- CONFSEND
- FREE
- PENDFREE
- PENDRECEIVE
- RECEIVE
- ROLLBACK
- SEND
- SYNCFREE
- SYNCRECEIVE
- SYNCSEND

Bedingungen

16 INVREQ

RESP2-Werte:

Eine Distributed Program Link-Serveranwendung hat die Funktionsverlagerungssitzung (ihre Hauptfunktion) explizit in der Option CONVID oder implizit durch den Standardwert angegeben.

- Der Befehl wurde für einen CPI Communications-Dialog abgesetzt.
- Der Befehl wurde für einen APPC-Basisdialog abgesetzt. (Stattdessen hätte ein Befehl GDS EXT-RACT ATTRIBUTES verwendet werden sollen.)

Tritt auf, wenn sich der in CONVID angegebene Wert nicht auf einen Dialog bezieht, dessen Eigner die Anwendung ist.

Kapitel 2. Zusammenfassung der CICS-Befehle **177**

- SEND
- SYNCFREE
- SYNCRECEIVE
- SYNCSEND

Bedingungen

16 INVREQ

RESP2-Werte:

200

Eine Distributed Program Link-Serveranwendung hat die Funktionsverlagerungssitzung (ihre Hauptfunktion) explizit in der Option CONVID oder implizit durch den Standardwert angegeben.

Tritt außerdem in jedem der folgenden Fälle auf (RESP2 nicht festgelegt):

- Ein falscher Befehl wurde für das verwendete Terminal oder die verwendete logische Einheit (LU) abgesetzt.

Standardaktion: Die Task wird abnormal beendet.

61 NOTALLOC

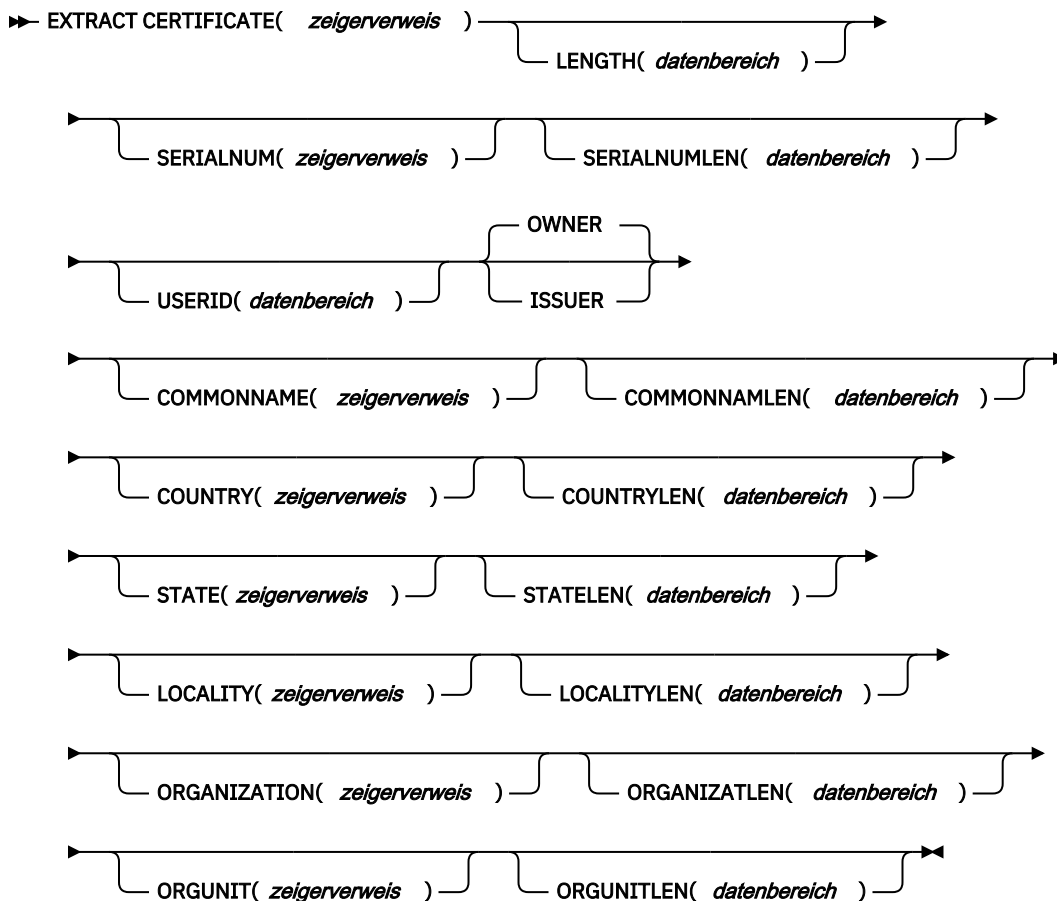
Tritt auf, wenn die im Befehl angegebene Funktion (Facility) nicht der Anwendung gehört.

Standardaktion: Die Task wird abnormal beendet.

EXTRACT CERTIFICATE

Ruft Informationen aus dem Clientzertifikat ab, das durch einen TCP/IP-Service unter Angabe der Clientauthentifizierung empfangen wurde.

EXTRACT CERTIFICATE



Bedingung: INVREQ

Dieser Befehl ist threadsicher.

Beschreibung

Mit dem Befehl EXTRACT CERTIFICATE kann die Anwendung Informationen aus dem X.509-Zertifikat abrufen, das von einem Client bei einem SSL-Handshake (Secure Sockets Layer) durch einen TCP/IP-Service (TCPIPSERVICE) empfangen wurde, für den SSL(CLIENTAUTH) angegeben war. Das Zertifikat enthält Felder, die den Eigner (oder das Subjekt) des Zertifikats angeben, sowie Felder, die die Zertifizierungsstelle (CA) angeben, die das Zertifikat ausgestellt hat. Sie können die Felder, die Sie benötigen, mit der Option OWNER oder mit der Option ISSUER auswählen. Sie können jedoch nicht die Felder für OWNER und ISSUER in einem Befehl abrufen.

Wenn Sie versuchen, ein Zertifikat zu extrahieren und kein Zertifikat zum Extrahieren vorhanden ist, werden niedrige Werte oder Nullwerte für die Zeiger zurückgegeben.

Optionen

CERTIFICATE(zeigerverweis)

Gibt einen Zeigerverweis an, der auf die Adresse des Zertifikats, das vom Client empfangen wurde, als Vollwort-Binärwert gesetzt werden soll. Der Zeigerverweis ist bis zum nächsten CICS-Befehl oder bis zum Ende der Task gültig.

COMMONNAME(zeigerverweis)

Gibt einen Zeigerverweis an, der auf den allgemeinen Namen aus dem Clientzertifikat gesetzt werden soll. Der Zeigerverweis ist bis zum nächsten CICS-Befehl oder bis zum Ende der Task gültig.

COMMONNAMLEN(datenbereich)

Gibt einen Vollwort-Binärwert-Datenbereich an, der auf die Länge des allgemeinen Namens aus dem Clientzertifikat gesetzt werden soll.

COUNTRY(zeigerverweis)

Gibt einen Zeigerverweis an, der auf die Adresse des Landes aus dem Clientzertifikat gesetzt werden soll. Der Zeigerverweis ist bis zum nächsten CICS-Befehl oder bis zum Ende der Task gültig.

COUNTRYLEN(datenbereich)

Gibt einen Vollwort-Binärwert-Datenbereich an, der auf die Länge des Landes aus dem Clientzertifikat gesetzt werden soll.

ISSUER

Gibt an, dass die Werte, die durch diesen Befehl zurückgegeben werden, zu der Zertifizierungsstelle gehören, die dieses Zertifikat ausgestellt hat.

LENGTH(datenbereich)

Gibt einen Vollwort-Binärwert-Datenbereich an, der auf die Länge des Hauptteils des Clientzertifikats gesetzt werden soll.

LOCALITY(zeigerverweis)

Gibt einen Zeigerverweis an, der auf die Adresse des Standorts aus dem Clientzertifikat gesetzt werden soll. Der Zeigerverweis ist bis zum nächsten CICS-Befehl oder bis zum Ende der Task gültig.

LOCALITYLEN(datenbereich)

Gibt einen Vollwort-Binärwert-Datenbereich an, der auf die Länge des Standorts aus dem Clientzertifikat gesetzt werden soll.

ORGANIZATION(zeigerverweis)

Gibt einen Zeigerverweis an, der auf die Adresse der Organisation aus dem Clientzertifikat gesetzt werden soll. Der Zeigerverweis ist bis zum nächsten CICS-Befehl oder bis zum Ende der Task gültig.

ORGANIZATLEN(datenbereich)

Gibt einen Vollwort-Binärwert-Datenbereich an, der auf die Länge der Organisation aus dem Clientzertifikat gesetzt werden soll.

ORGUNIT(zeigerverweis)

Gibt einen Zeigerverweis an, der auf die Adresse der Organisationseinheit aus dem Clientzertifikat gesetzt werden soll. Der Zeigerverweis ist bis zum nächsten CICS-Befehl oder bis zum Ende der Task gültig.

ORGUNITLEN(datenbereich)

Gibt einen Vollwort-Binärwert-Datenbereich an, der auf die Länge der Organisationseinheit aus dem Clientzertifikat gesetzt werden soll.

OWNER

Gibt an, dass die Werte, die durch diesen Befehl zurückgegeben werden, zum Eigner des Zertifikats gehören.

SERIALNUM(zeigerverweis)

Gibt einen Zeigerverweis an, der auf die Adresse der Seriennummer des Zertifikats, die vom Zertifikatsaussteller zugewiesen wurde, gesetzt werden soll. Der Zeigerverweis ist bis zum nächsten CICS-Befehl oder bis zum Ende der Task gültig.

SERIALNUMLEN(datenbereich)

Gibt einen Vollwort-Binärwert-Datenbereich an, der auf die Länge der Seriennummer gesetzt werden soll.

STATE(zeigerverweis)

Gibt einen Zeigerverweis an, der auf die Adresse des Staates oder Bundeslands aus dem Clientzertifikat gesetzt werden soll. Der Zeigerverweis ist bis zum nächsten CICS-Befehl oder bis zum Ende der Task gültig.

STATELEN(datenbereich)

Gibt einen Vollwort-Binärwert-Datenbereich an, der auf die Länge des Staates oder Bundeslands aus dem Clientzertifikat gesetzt werden soll.

USERID(datenbereich)

Gibt ein acht Byte langes Feld an, das auf die Benutzer-ID gesetzt werden soll, die mit dem Clientzertifikat verbunden ist.

Bedingungen

16 INVREQ

RESP2-Werte:

5

Der Befehl wurde aus einer Nicht-TCPIP-Anwendung abgesetzt.

EXTRACT LOGONMSG

Greift auf Anmeldedaten von z/OS Communications Server zu.

EXTRACT LOGONMSG

```

➤ EXTRACT LOGONMSG      └── INTO( datenbereich ) ── LENGTH( datenbereich ) ➤
                        └── SET( zeigerverweis ) ──

```

Bedingung: NOTALLOC

Beschreibung

Der Befehl EXTRACT LOGONMSG greift auf die Anmeldedaten von z/OS Communications Server zu. Diese Daten können zum Beispiel vom Terminalbediener bei der Anmeldung oder in einem Befehl ISSUE PASS angegeben worden sein. Diese Daten sind nur verfügbar, wenn der Systeminitialisierungsparameter LGNMSG=YES angegeben wurde. Die Daten können nur einmal extrahiert werden. Mithilfe des Systeminitialisierungsparameters GMTRAN kann erzwungen werden, dass die erste Transaktion, die auf dem Terminal ausgeführt wird, diejenige ist, die den Befehl EXTRACT LOGONMSG absetzt.

Alle Anmeldedaten werden extrahiert und die Länge der Daten wird in das Feld eingefügt, das in der Option LENGTH angegeben wird. Da die Option LENGTH nicht verwendet werden kann, um die Menge der extrahierten Daten zu begrenzen, wird empfohlen, immer ein 256 Byte langes Feld für diese Option zu verwenden.

Wenn Sie die Option SET verwenden, werden z/OS Communications Server-Anmeldedaten erst freigegeben, wenn die Sitzung beendet wird (CLSDST). Wenn Sie die Option INTO verwenden, werden die z/OS Communications Server-Anmeldedaten in den Benutzerspeicher kopiert und anschließend freigegeben.

Optionen

INTO(*datenbereich*)

Gibt das empfangende Feld für die extrahierten Daten an.

LENGTH(*datenbereich*)

Gibt die Länge der extrahierten Daten als Halbwort-Binärwert an. Wenn keine Daten verfügbar sind, wird LENGTH auf null gesetzt.

SET(zeigerverweis)

Gibt den Zeigerverweis an, der auf die Adresse der extrahierten Daten gesetzt werden soll. Der Zeigerverweis bleibt bis zum nächsten Befehl `EXTRACT LOGONMSG` oder bis zum Ende der Task gültig, sofern er nicht durch andere Befehle oder Anweisungen geändert wird.

Wenn dem Anwendungsprogramm `DATALOCATION(ANY)` zugeordnet ist, kann die Adresse der Daten ober- oder unterhalb der 16-MB-Grenze liegen.

Wenn dem Anwendungsprogramm DATALOCATION(BELOW) zugeordnet ist und sich die Daten oberhalb der 16-MB-Grenze befinden, werden die Daten unter die 16-MB-Grenze kopiert und die Adresse dieser Kopie wird zurückgegeben.

Wenn TASKDATAKEY(USER) für die aktive Task angegeben wird und der Speicherschutz aktiv ist, werden die Daten in einem Benutzerschlüssel zurückgegeben. Wenn für die aktive Task TASKDATAKEY(CICS) angegeben wird und der Speicherschutz aktiv ist, befinden sich die zurückgegebenen Daten in einem CICS-Schlüssel.

Bedingungen

61 NOTALLOC

Tritt auf, falls die Anwendung nicht Eigner der im Befehl angegebenen Funktion ist.

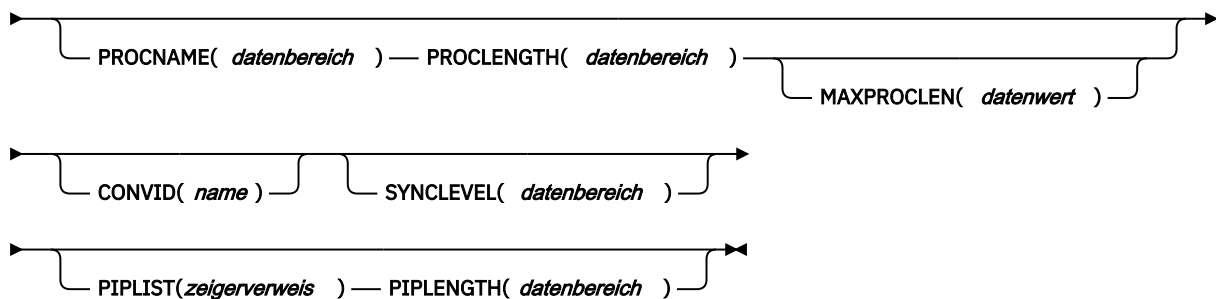
Standardaktion: Die Task wird abnormal beendet.

EXTRACT PROCESS

Ruft Werte aus einem Anschlussheader eines APPC-Dialogs ab.

EXTRACT PROCESS (APPC)

▶▶ **EXTRACT PROCESS** ▶▶



Bedingungen: INVREQ, LENGERR, NOTALLOC

Beschreibung

Mit dem Befehl `EXTRACT PROCESS` kann eine Anwendung auf dialogbezogene Daten zugreifen, die für CICS angegeben werden, wenn das Programm angeschlossen (`ATTACH`) wird. Der Anschlussempfänger muss den Befehl `EXTRACT PROCESS` nur ausführen, wenn er diese Informationen benötigt.

Der Befehl **EXTRACT PROCESS** ist nur in einem APPC-Dialog gültig, der die Hauptfunktion für die Task ist.

Optionen

CONVID(*name*)

Gibt den Dialog an, auf den sich der Befehl bezieht. Der vier Zeichen lange Name identifiziert das Token, das die Hauptsitzung (EIBTRMID) darstellt.

Aus Gründen der Kompatibilität mit früheren Releases wird SESSION als Synonym für CONVID akzeptiert. Neue Programme sollten CONVID verwenden.

Wenn weder CONVID noch SESSION angegeben wird, wird standardmäßig die Hauptfunktion für die Task verwendet.

MAXPROCLEN(datenwert)

Gibt die Pufferlänge für PROCNAME an. Wenn die Option MAXPROCLEN nicht angegeben wird, wird eine Pufferlänge von 32 Byte angenommen.

PIPLENGTH(datenbereich)

Gibt einen Halbwort-Binärwert-Datenbereich an, in dem die Gesamtlänge der Liste der Prozessinitialisierungsparameter (PIP) zurückgegeben wird.

PIPLIST(zeigerverweis)

Gibt einen Zeigerverweis an, der auf die Adresse eines von CICS bereitgestellten Datenbereichs gesetzt wird, der eine PIP-Liste enthält. Diese Liste enthält Datensätze variabler Länge in demselben Format wie die Liste im Befehl CONNECT PROCESS. Die Rückgabe des Werts null bedeutet, dass keine PIP-Daten von CICS empfangen wurden.

PROCLNGTH(datenbereich)

Gibt einen Halbwort-Datenbereich an, der von CICS auf die Länge des Prozessnamens gesetzt wird. Wenn PROCNAME angegeben wird, muss diese Option angegeben werden.

PROCNAME(datenbereich)

Gibt den Datenbereich an, der den Prozessnamen empfangen soll, der durch das ferne System angegeben wird, das den Start der Task veranlasst hat. Der Datenbereich kann 1 – 64 Byte lang sein. Der Prozessname wird auf der rechten Seite mit Leerzeichen aufgefüllt, wenn er zu kurz ist. Der Datenbereich PROCNAME sollte nicht kürzer als der Wert in MAXPROCLEN sein.

SYNCLEVEL(datenbereich)

Gibt einen Halbwort-Datenbereich an, der von CICS auf den Wert von SYNCLEVEL (Synchronisationsebene) gesetzt wird. Weitere Informationen zu Synchronisationsebenen finden Sie unter [Synchronisation](#).

Bedingungen**16 INVREQ**

RESP2-Werte:

200

Eine Serveranwendung für Distributed Program Link hat in der Option CONVID die Sitzung für die Funktionsverlagerung (ihre Hauptfunktion) angegeben.

Tritt außerdem in einer der folgenden Situationen auf (RESP2 nicht festgelegt):

- Der Befehl EXTRACT PROCESS wurde für einen anderen Dialog als einen formatfreien APPC-Dialog (z. B. LUTYPE6.1-Dialog, APPC-Basisdialog oder CPI Communications) verwendet.
- Der Befehl EXTRACT PROCESS wurde für einen Dialog verwendet, der nicht durch Eingaben aus dem Netz gestartet wurde und dessen Sitzung keine Hauptfunktion ist.
- Der Befehl wurde für einen CPI Communications-Dialog abgesetzt.

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

Tritt auf, wenn die tatsächliche Länge von PROCNAME größer als MAXPROCLEN ist oder größer als 32 Byte ist, MAXPROCLEN nicht angegeben wurde.

Standardaktion: Die Task wird abnormal beendet.

61 NOTALLOC

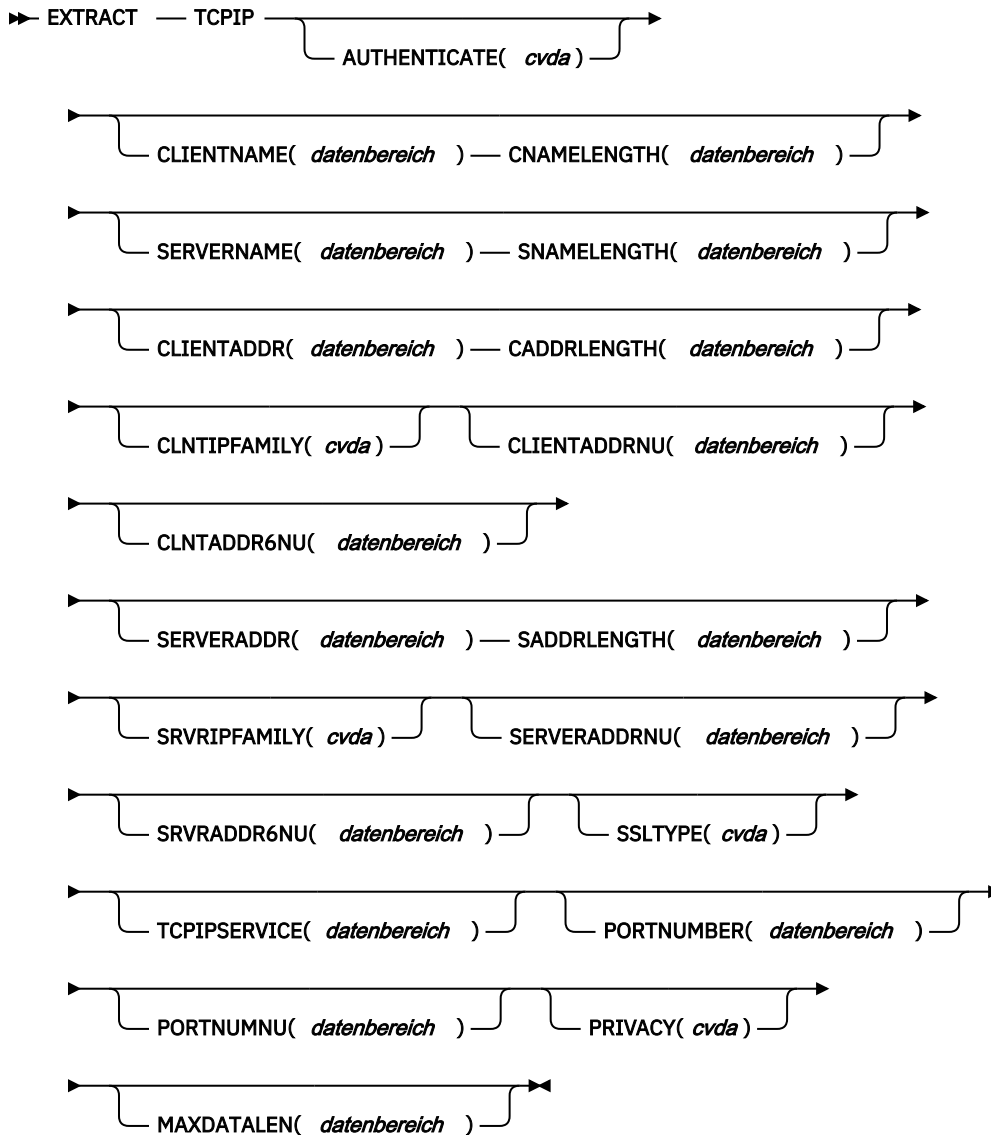
Tritt auf, wenn sich der in CONVID angegebene Wert nicht auf einen Dialog bezieht, dessen Eigner die Anwendung ist.

Standardaktion: Die Task wird abnormal beendet.

EXTRACT TCPIP

Ruft Informationen zu den TCP/IP-Merkmalen der aktuellen Transaktion ab.

EXTRACT TCPIP



Bedingungen: INVREQ, LENGERR

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl **EXTRACT TCPIP** stellt Informationen zur TCP/IP-Verbindung und zu den in der **TCPIPSERVICE**-Definition angegebenen Sicherheitsoptionen bereit.

Optionen

AUTHENTICATE(cvda)

Gibt einen CVDA-Wert zurück, der die Authentifizierung angibt, die für den Client mit dieser Transaktion angefordert wurde. Die folgenden Werte sind möglich:

ASSERTED

AUTOAUTH

AUTOREGISTER

BASICAUTH

CERTIFICAUTH

NOAUTHENTIC

CADDRLENGTH(datenbereich)

Gibt die Länge des Puffers zurück, der in der Option CLIENTADDR angegeben wird, und wird auf die Länge der Daten gesetzt, die an die Anwendung zurückgegeben werden. Wenn die Option CLIENTADDR eine IPv6-Adresse angibt, müssen Sie die Pufferlänge in CADDRLENGTH mindestens auf 39 Zeichen setzen. Wenn die Daten die Pufferlänge überschreiten, wird eine Bedingung LENGERR ausgelöst und die Daten werden abgeschnitten.

CLIENTADDR(datenbereich)

Gibt einen Puffer zurück, der die IP-Adresse des Clients enthält. Die IP-Adresse kann das IPv4- oder das IPv6-Format haben. IPv4-Adressen werden in Form nativer IPv4-Adressen in der Schreibweise mit Trennzeichen zurückgegeben. Beispiel: 1.2.3.4. IPv6-Adressen werden in Form nativer IPv6-Adressen in Hexadezimalschreibweise mit Doppelpunkten zurückgegeben. Beispiel: ::a:b:c:d

Weitere Informationen zu IP-Adressen finden Sie unter [IP-Adressen](#).

CLIENTADDRNU(datenbereich)

Gibt ein Vollwort-Binärfeld zurück, das die IPv4-Adresse des Clients im Binärformat enthält. Wenn die Adresse das IPv6-Format hat, wird sie in der Option CLNTADDR6NU zurückgegeben und in der Option CLIENTADDRNU wird 0 zurückgegeben.

CLIENTNAME(datenbereich)

Gibt einen Puffer an, der den Namen des Clients enthalten soll, wie er dem Domänennamensserver (DNS) bekannt ist.

CLNTADDR6NU(datenbereich)

Gibt ein 16 Byte langes Feld zurück, das die IPv6-Adresse des Clients im Binärformat enthält. Diese Option wird nur zurückgegeben, wenn die Option CLNTIPFAMILY den Wert IPV6 hat. Wenn Adresse das IPv4-Format hat, wird die Adresse in der Option CLNTADDRNU zurückgegeben und in der Option CLNTADDR6NU werden Nullen zurückgegeben.

CLNTIPFAMILY(cvda)

Gibt das Format der IP-Adresse des Clients zurück. Die CVDA-Werte lauten wie folgt:

IPV4

CLIENTADDR gibt eine IPv4-Adresse in der Schreibweise mit Trennzeichen zurück und CLIENTADDRNU gibt die IPv4-Adresse im Binärformat zurück.

IPV6

CLIENTADDR gibt eine hexadezimale IPv6-Adresse in Doppelpunktschreibweise zurück und CLIENTADDR6NU gibt die IPv6-Adresse im Binärformat zurück.

NOTAPPLIC

Die Quelle der Eingabe wurde nicht ermittelt. Es wird der Wert 0.0.0.0 zurückgegeben.

CNAMELENGTH(datenbereich)

Gibt die Länge des Puffers an, der in der Option CLIENTNAME angegeben wird, und wird auf die tatsächliche Länge der an die Anwendung zurückgegebenen Daten gesetzt oder auf null, wenn der Name des Clients dem Domänennamensserver nicht bekannt ist. Falls die Daten die Puffergröße überschreiten, tritt eine Bedingung LENGERR auf und die Daten werden abgeschnitten.

MAXDATALEN(datenbereich)

Gibt ein Vollwort-Binärfeld an, das die Einstellung für die maximale Länge der Daten enthalten soll, die von CICS als HTTP-Server empfangen werden können.

PRIVACY(cvda)

Gibt einen CVDA-Wert zurück, der die Ebene der SSL-Verschlüsselung angibt, die zwischen der Transaktion und ihrem Client für eine eingehende Anforderung verwendet wird. Die folgenden CVDA-Werte sind möglich:

NOTSUPPORTED

REQUIRED

SUPPORTED

PORTNUMBER(datenbereich)

Gibt ein fünf Zeichen langes Feld im Zeichenformat an, das die Portnummer enthalten soll, die dieser Transaktion zugeordnet ist. Dieser Port hat die eingehenden Daten empfangen, die diese Transaktion initialisiert haben.

PORTNUMNU(datenbereich)

Gibt ein Vollwortfeld im Binärformat an, das die Portnummer enthalten soll, die dieser Transaktion zugeordnet ist. Dieser Port hat die eingehenden Daten empfangen, die diese Transaktion initialisiert haben.

SADDRLLENGTH(datenbereich)

Gibt die Länge des Puffers zurück, der in der Option SERVERADDR angegeben wird, und wird auf die Länge der Daten gesetzt, die an die Anwendung zurückgegeben werden. Wenn SERVERADDR eine IPv6-Adresse angibt, müssen Sie die Pufferlänge in SADDRLLENGTH auf mindestens 39 Zeichen setzen. Wenn die Daten die Pufferlänge überschreiten, wird eine Bedingung LENGERR ausgelöst und die Daten werden abgeschnitten.

SERVERADDR(datenbereich)

Gibt einen Puffer zurück, der die IP-Adresse des Servers enthält. Die IP-Adresse kann das IPv4- oder das IPv6-Format haben. IPv4-Adressen werden in Form nativer IPv4-Adressen in der Schreibweise mit Trennzeichen zurückgegeben. Beispiel: 1 . 2 . 3 . 4. IPv6-Adressen werden in Form nativer IPv6-Adressen in Hexadezimalschreibweise mit Doppelpunkten zurückgegeben. Beispiel: : : a : b : c : d. Wenn ein Fehler auftritt, wird der Wert 0 . 0 . 0 . 0 zurückgegeben und die Daten werden abgeschnitten.

SERVERADDRNU(datenbereich)

Gibt ein Vollwort-Binärfeld zurück, das die IPv4-Adresse des Servers im Binärformat enthält. Wenn die Adresse das IPv6-Format hat, wird sie in der Option SRVADDR6NU zurückgegeben und in der Option SERVERADDRNU wird 0 zurückgegeben.

SERVERNAME(datenbereich)

Gibt einen Puffer an, der den Namen des Servers enthalten soll, wie er dem Domännennamensserver (DNS) bekannt ist.

SNAMELENGTH(datenbereich)

Gibt die Länge des Puffers an, der in der Option SERVERNAME angegeben wird, und wird auf die Länge der Daten gesetzt, die an die Anwendung zurückgegeben werden. Wenn die Daten die Pufferlänge überschreiten, wird eine Bedingung LENGERR ausgelöst und die Daten werden abgeschnitten.

SRVADDR6NU(datenbereich)

Gibt ein 16 Byte langes Feld zurück, das die IPv6-Adresse des Servers im Binärformat enthält. Diese Option wird nur zurückgegeben, wenn die Option SRVIPFAMILY den Wert IPV6 hat. Wenn Adresse das IPv4-Format hat, wird die Adresse in der Option SERVERADDRNU zurückgegeben und in der Option SRVADDR6NU werden Nullen zurückgegeben.

SRVIPFAMILY(cvda)

Gibt das Format der IP-Adresse des Servers zurück. Die CVDA-Werte lauten wie folgt:

IPV4

SERVERADDR gibt eine IPv4-Adresse in der Schreibweise mit Trennzeichen zurück und SERVERADDRNU gibt die IPv4-Adresse im Binärformat zurück.

IPV6

SERVERADDR gibt eine hexadezimale IPv6-Adresse in Doppelpunktschreibweise zurück und SERVERADDR6NU gibt die IPv6-Adresse im Binärformat zurück.

NOTAPPLIC

Die Quelle der Eingabe wurde nicht ermittelt. Es wird der Wert 0 . 0 . 0 . 0 zurückgegeben.

SSLTYPE(cvda)

Gibt einen CVDA-Wert zurück, der angibt, ob Secure Sockets Layer (SSL) verwendet wird, um die Kommunikation für diese Transaktion zu schützen. Die folgenden Werte sind möglich:

SSL

NOSSL

CLIENTAUTH

ATTLSAWARE

TCPIPSERVICE(*datenbereich*)

Ein acht Byte langes Feld, das den Namen des TCP/IP-Service enthalten soll, der dieser Transaktion zugeordnet ist.

Bedingungen

16 INVREQ

RESP2-Werte:

2

Es wurde eine falsche Socketantwort empfangen.

5

Der Befehl wurde aus einer Nicht-TCP/IP-Anwendung abgesetzt.

22 LENGERR

RESP2-Werte:

1

CLIENTADDR, SERVERADDR, CLIENTNAME oder SERVERNAME wurde angegeben, jedoch wurde das relevante Längenfeld entweder nicht angegeben oder es enthält einen Wert kleiner oder gleich null.

3

CLIENTADDR ist zu klein, um die extrahierte Zeichenfolge aufzunehmen.

4

SERVERADDR ist zu klein, um die extrahierte Zeichenfolge aufzunehmen.

6

CLIENTNAME ist zu klein, um die extrahierte Zeichenfolge aufzunehmen.

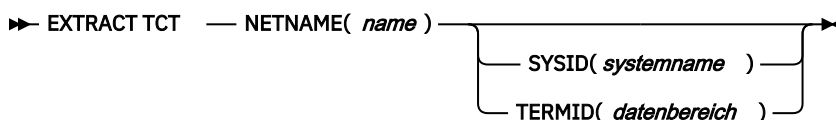
7

SERVERNAME ist zu klein, um die extrahierte Zeichenfolge aufzunehmen.

EXTRACT TCT

Konvertiert einen acht Zeichen langen Namen in einen vier Zeichen langen Namen auf einer logische Einheit vom Typ LU-Typ 6.1 (LUTYPE6.1).

EXTRACT TCT



Bedingungen: INVREQ, NOTALLOC

Beschreibung

Der Befehl EXTRACT TCT konvertiert den acht Zeichen langen SNA-Netznamen für eine logische Einheit in den entsprechenden vier Zeichen langen Namen, der im lokalen CICS-System bekannt ist.

Optionen

NETNAME(*name*)

Gibt den acht Zeichen langen Namen der logischen Einheit im SNA-Netz an.

SYSID(*systemname*)

Gibt die Variable an, die auf den äquivalenten lokalen Namen des Systems gesetzt werden soll.

TERMID(*datenbereich*)

Gibt die Variable an, die auf den äquivalenten lokalen Namen des Terminals gesetzt werden soll.

Bedingungen

16 INVREQ

Tritt auf, wenn NETNAME nicht gültig ist.

Standardaktion: Die Task wird abnormal beendet.

61 NOTALLOC

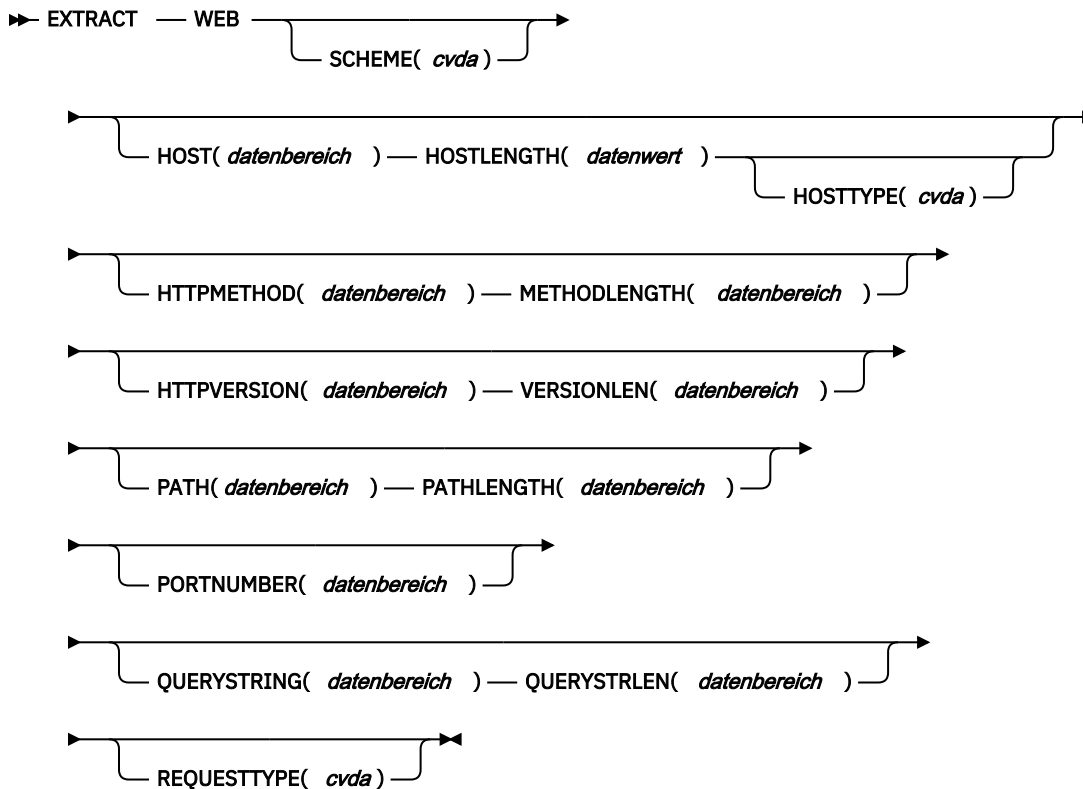
Tritt auf, falls die Anwendung nicht Eigner der im Befehl angegebenen Funktion ist.

Standardaktion: Die Task wird abnormal beendet.

EXTRACT WEB

Ruft Informationen zu einer HTTP-Anforderung, die an CICS als HTTP-Server ausgegeben wurde, oder zu einer Verbindung zwischen einem Internet-Server und CICS als HTTP-Client ab. Dieser Befehl ist ein Synonym für WEB EXTRACT.

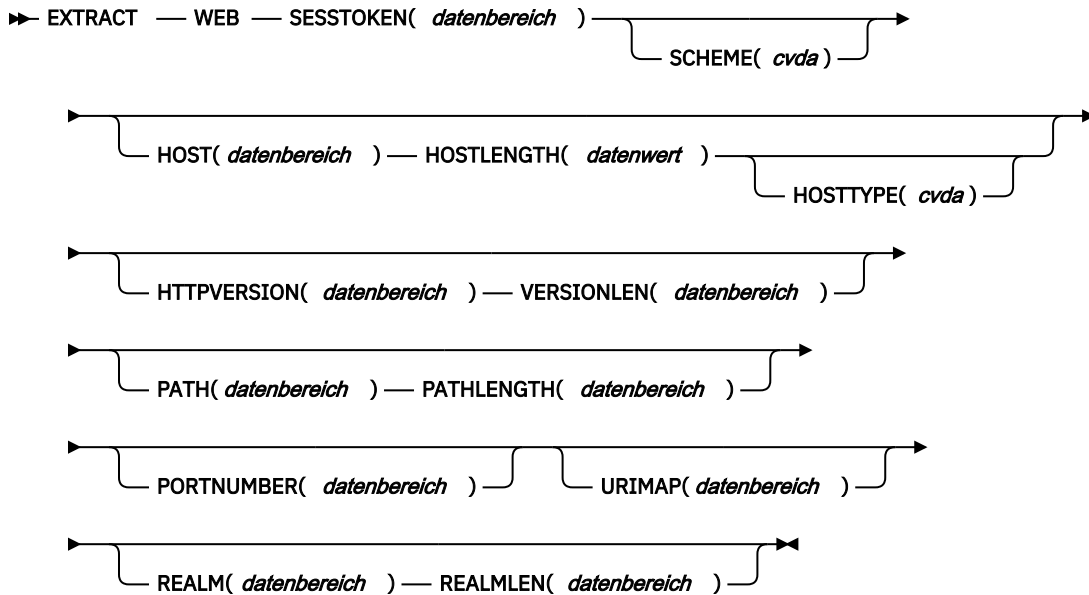
EXTRACT WEB (CICS als HTTP-Server)



Bedingungen: INVREQ, LENGERR, NOTOPEN

Dieser Befehl ist threadsicher.

EXTRACT WEB (CICS als HTTP-Client)



Bedingungen: INVREQ, IOERR, LENGERR, NOTFND, NOTOPEN, TIMEDOUT

Dieser Befehl ist threadsicher.

Beschreibung

Wenn CICS als HTTP-Server eingesetzt wird, kann eine Anwendung mit dem Befehl **WEB EXTRACT** Informationen zur letzten HTTP-Anforderung abrufen, die durch einen Web-Client an CICS ausgegeben und der Anwendung zur Verarbeitung zugeordnet wurde.

Wenn CICS als HTTP-Client eingesetzt wird und die Option **SESSTOKEN** angegeben ist, kann eine Anwendung mit dem Befehl Informationen zu einer Verbindung abrufen, die sie mit einem Server geöffnet hat. Die Informationen, die an die Anwendung zurückgegeben werden, beinhalten globale Angaben über die Verbindung wie beispielsweise den Hostnamen und die HTTP-Version des Servers. Informationen zu bestimmten Anforderungen, die durch die Anwendung ausgegeben wurden, und zu Antworten des Servers werden bei Verwendung dieses Befehls nicht zur Verfügung gestellt. Zum Abrufen von Informationen aus einer Serverantwort wird der Befehl **WEB RECEIVE** verwendet.

Optionen

HOST(datenbereich)

Wenn CICS als HTTP-Server eingesetzt wird, gibt die Option **HOST** einen Puffer an, der die Hostkomponente der URL aufnehmen soll, die entweder im Headerfeld 'Host' für die Anforderung oder in der Anforderungszeile angegeben ist (falls für die Anforderung eine absolute URI verwendet wurde). Die Portnummer wird separat mit der Option **PORTNUMBER** dargestellt.

Wenn CICS als HTTP-Client eingesetzt wird und die Option **SESSTOKEN** verwendet wird, gibt die Option **HOST** einen Puffer an, der den Hostnamen des Servers in der Verbindung aufnehmen soll, die durch die Option **SESSTOKEN** gekennzeichnet ist. Die Portnummer wird separat mit der Option **PORTNUMBER** dargestellt.

Der Hostname kann durch eine IPv4- oder eine IPv6-Adresse dargestellt werden. IPv4-Adressen werden als native IPv4-Adressen in der Schreibweise mit Trennzeichen zurückgegeben. Beispiel: 1.2.3.4. IPv6-Adressen werden als native IPv6-Hexadezimaladressen mit Doppelpunkten zurückgegeben. Beispiel: ::a:b:c:d.

Informationen zu IP-Adressen finden Sie unter [IP-Adressen](#).

HOSTLENGTH(datenbereich)

Gibt als Vollwort-Binärzahlvariable die Länge des in der Option HOST angegebenen Puffers an und wird auf die Länge der Daten gesetzt, die an die Anwendung zurückgegeben werden. Als Angabe für diesen Datenbereich ist eine Größe von 116 Zeichen zweckmäßig. Falls die Daten die Puffergröße überschreiten, tritt eine Bedingung LENGERR auf und die Daten werden abgeschnitten.

HOSTTYPE(cvda)

Gibt das Adressformat für den Wert der Option HOST zurück. Die CVDA-Werte lauten wie folgt:

HOSTNAME

Die Option HOST enthält einen aus Zeichen bestehenden Hostnamen. Die IP-Adresse, die dem Hostnamen entspricht, wird im Domänennamensserver ermittelt.

IPV4

Die Adresse ist eine IPv4-Adresse in der Schreibweise mit Trennzeichen.

IPV6

Die Adresse ist eine IPv6-Hexadezimaladresse mit Doppelpunkten.

NOTAPPLIC

Es wurde eine falsche Hostadresse zurückgegeben (HOST=0.0.0.0).

HTTPMETHOD(datenbereich)

Wenn CICS als HTTP-Server eingesetzt wird, gibt diese Option einen Puffer an, der die HTTP-Methodezeichenfolge in der Anforderungszeile der Nachricht aufnehmen soll.

Beim Einsatz von CICS als HTTP-Client ist diese Option nicht relevant.

HTTPVERSION(datenbereich)

Wenn CICS als HTTP-Server eingesetzt wird, gibt diese Option einen Puffer an, der die in der Anforderung angegebene HTTP-Version für den Web-Client aufnehmen soll.

Wenn CICS als HTTP-Client eingesetzt wird und die Option SESSTOKEN angegeben ist, gibt diese Option einen Puffer an, der die HTTP-Version des Servers in der Verbindung aufnehmen soll, die durch die Option SESSTOKEN gekennzeichnet ist. Falls CICS die HTTP-Version des Servers nicht kennt, gibt CICS an den Server eine Anforderung mit der Methode OPTIONS aus, um diese Informationen zu ermitteln.

1.1 steht für HTTP/1.1; 1.0 steht für HTTP/1.0 und früher.

METHODLENGTH(datenbereich)

Gibt als Vollwort-Binärzahlvariable die Länge des in der Option HTTPMETHOD angegebenen Puffers an und wird auf die tatsächliche Länge der Daten gesetzt, die an die Anwendung zurückgegeben werden. Falls die Daten die Puffergröße überschreiten, tritt eine Bedingung LENGERR auf und die Daten werden abgeschnitten.

PATH(datenbereich)

Wenn CICS als HTTP-Server eingesetzt wird, gibt diese Option einen Puffer an, der den Pfad in der Anforderungszeile der Nachricht aufnehmen soll.

Wenn CICS als HTTP-Client eingesetzt wird und die Option SESSTOKEN angegeben ist, gibt diese Option einen Puffer an, der den gültigen Standardpfad für Anforderungen, die mithilfe der Verbindung ausgegeben werden, aufnehmen soll. Falls eine URIMAP-Definition im Befehl WEB OPEN für die Verbindung angegeben wurde, ist der Standardpfad derjenige Pfad, der in der URIMAP-Definition festgelegt ist. Andernfalls besteht der Standardpfad aus einem einzelnen normalen Schrägstrich.

PATHLENGTH(datenbereich)

Gibt als Vollwort-Binärzahlvariable die Länge des in der Option PATH angegebenen Puffers an und wird auf die Länge der Daten gesetzt, die an die Anwendung zurückgegeben werden. Als Angabe für diesen Datenbereich ist eine Größe von 256 Zeichen zweckmäßig. Falls die Daten die Puffergröße überschreiten, tritt eine Bedingung LENGERR auf und die Daten werden abgeschnitten.

PORTNUMBER(datenbereich)

Wenn CICS als HTTP-Server eingesetzt wird, gibt diese Option einen Datenbereich zurück, der die Portnummer enthält, die in der Anforderungszeile der Nachricht angegeben ist.

Wenn CICS als HTTP-Client eingesetzt wird und die Option SESSTOKEN angegeben ist, gibt diese Option einen Datenbereich zurück, der die Portnummer enthält, mit der auf den Server in der Verbindung zugegriffen wird, die durch die Option SESSTOKEN gekennzeichnet ist.

Der im Datenbereich zurückgegebene Wert ist ein Vollwort-Binärwert.

Standardisierte Portnummern für einen Service werden in der URL normalerweise nicht angegeben. Falls die Portnummer in der URL nicht vorhanden ist, wird die Nummer vom Befehl anhand des Schemas ermittelt und zurückgegeben. Die standardisierte Portnummer für HTTP ist 80; bei HTTPS lautet die standardisierte Portnummer 443. Falls eine Portnummer zurückgegeben wird, die nicht den Standardwert für das Schema darstellt, müssen Sie die Portnummer explizit angeben, um Zugriff auf die URL zu erhalten; dies gilt beispielsweise, wenn Sie diese Informationen in einem Befehl WEB OPEN verwenden.

QUERYSTRING(datenbereich)

Wenn CICS als HTTP-Server eingesetzt wird, gibt diese Option einen Puffer an, der die Abfragezeichenfolge in der Anforderungszeile der Nachricht aufnehmen soll. Die Abfragezeichenfolge besteht aus einem oder mehreren Werten, die nach dem Fragezeichen (?) codiert sind, mit dem das Ende des Pfades begrenzt wird. Die Abfragezeichenfolge wird in ihrem Format mit Escapezeichen zurückgegeben.

Beim Einsatz von CICS als HTTP-Client ist diese Option nicht relevant.

QUERYSTRLEN(datenbereich)

Gibt als Vollwort-Binärzahlvariable die Länge des in der Option QUERY angegebenen Puffers an und wird auf die Länge der Daten gesetzt, die an die Anwendung zurückgegeben werden (die Abfragezeichenfolge). Als Angabe für diesen Datenbereich ist eine Größe von 256 Zeichen zweckmäßig. Falls die Daten die Puffergröße überschreiten, tritt eine Bedingung LENGERR auf und die Daten werden abgeschnitten.

REALM(datenbereich)

Wenn CICS als HTTP-Client eingesetzt wird, gibt diese Option den Realm oder die Sicherheitsumgebung an, in dem/der die von Ihnen angeforderten Daten enthalten sind. Falls Sie einen Befehl als Antwort auf eine HTTP-Nachricht 401 ausgeben, gibt die Option REALM den Realmwert im zuletzt empfangenen Header 'WWW-Authenticate' an.

REALMLEN(datenbereich)

Wenn CICS als HTTP-Client eingesetzt wird, gibt diese Option als Vollwort-Binärzahlvariable die Länge des Puffers an, der für die Option REALM angegeben wurde. Falls Sie einen Befehl als Antwort auf eine HTTP-Nachricht 401 ausgeben, gibt die Option REALMLEN die Länge des Realmnamens im zuletzt empfangenen Header 'WWW-Authenticate' an.

REQUESTTYPE(cvda)

Wenn CICS als HTTP-Server eingesetzt wird, gibt diese Option den Typ der empfangenen Anforderung an. Beim Einsatz von CICS als HTTP-Client ist diese Option nicht relevant. Die CVDA-Werte lauten wie folgt:

HTTPYES

Gibt an, dass es sich um eine HTTP-Anforderung handelt.

HTTPNO

Gibt an, dass es sich bei der Anforderung nicht um eine HTTP-Anforderung handelt.

SCHEME(cvda)

Diese Option gibt sowohl bei Verwendung von CICS als HTTP-Server als auch beim Einsatz von CICS als HTTP-Client (mit angegebener Option SESSTOKEN) das Schema zurück, das für die Verbindung zwischen CICS und dem Web-Client oder Web-Server verwendet wird. Die CVDA-Werte lauten wie folgt:

HTTP

Gibt an, dass das HTTP-Protokoll ohne SSL verwendet wird.

HTTPS

Gibt an, dass das HTTPS-Protokoll, also HTTP mit SSL verwendet wird.

SESSTOKEN(datenwert)

Wenn CICS als HTTP-Client eingesetzt wird, ist diese Option erforderlich. Sie gibt den 8 Byte umfassenden Binärwert für das Sitzungstoken an, das eine Verbindung zwischen CICS und einem Server eindeutig kennzeichnet. Dieser Wert wird durch einen Befehl WEB OPEN für CICS als HTTP-Client zurückgegeben. Die Verwendung des Sitzungstokens ist im Abschnitt Sitzungstokens erläutert. Für den Befehl werden Informationen zur angegebenen Verbindung zurückgegeben.

Beim Einsatz von CICS als HTTP-Server ist diese Option nicht relevant.

URIMAP(datenbereich)

Wenn CICS als HTTP-Client eingesetzt wird und die Option SESSTOKEN angegeben ist, gibt diese Option den 8 Zeichen umfassenden Namen (in Groß-/Kleinschreibung) einer URIMAP-Definition zurück, die gegebenenfalls im Befehl WEB OPEN zum Öffnen der Verbindung angegeben wurde, die durch die Option SESSTOKEN gekennzeichnet ist. Mit dem Befehl INQUIRE URIMAP können Informationen zu den Attributen dieser URIMAP-Definition ermittelt werden.

Beim Einsatz von CICS als HTTP-Server ist diese Option nicht relevant.

VERSIONLEN(datenbereich)

Gibt als Vollwort-Binärzahlvariable die Länge des in der Option HTTPVERSION angegebenen Puffers an und wird auf die Länge der Daten gesetzt, die an die Anwendung zurückgegeben werden.

Bedingungen**16 INVREQ**

RESP2-Werte:

1

Der Befehl wurde in einer Anwendung ausgegeben, die nicht zur CICS-Webunterstützung gehört.

3

Der Befehl wurde für eine Anforderung ausgegeben, die keine HTTP-Anforderung ist. (Dieser Befehl wird nur festgelegt, wenn eine oder mehrere der Optionen HTTPMETHOD, HTTPVERSION oder PATH angegeben sind und es sich bei der Anforderung nicht um eine HTTP-Anforderung handelt).

41

Die Verbindung wurde geschlossen. Dies ist ein Fehler, der nur bei Verwendung des Befehls WEB EXTRACT für einen Client auftritt. Der Server hat möglicherweise aufgrund von Inaktivität über diese Verbindung ein zulässiges Zeitlimit überschritten.

67

Der Inhalt der Antwort entspricht nicht dem HTTP-Format. Der Fehler wird generiert, weil ein Syntaxproblem vorliegt. Dieser Fehler tritt nur bei Verwendung des Befehls WEB EXTRACT für einen Client auf.

71

Beim Chunked Transfer Coding ist ein Fehler aufgetreten. Dieser Fehler tritt nur bei Verwendung des Befehls WEB EXTRACT für einen Client auf.

144

Einer oder mehrere Parameter des Webbefehls sind ungültig. Dieser Fehler tritt nur bei Verwendung des Befehls WEB EXTRACT für einen Client auf.

17 IOERR

RESP2-Werte:

42

Es trat ein Socketfehler auf.

22 LENGERR

RESP2-Werte:

4

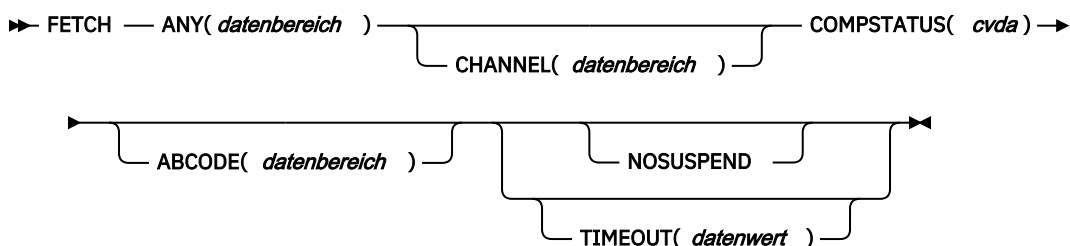
Die Methode überschreitet die (in der Option METHODLENGTH) angegebene Länge.

- 5**
Der Wert für die Option PATHLENGTH war nicht größer als Null.
- 6**
Die HTTP-Version überschreitet die (in der Option VERSIONLEN) angegebene Länge.
- 7**
Der Wert für die Option VERSIONLEN war nicht größer als Null.
- 8**
Die Abfragezeichenfolge überschreitet die (in der Option QUERYSTRLEN) angegebene Länge.
- 21**
Der Wert für die Option HOSTLENGTH war nicht größer als Null.
- 29**
Der Hostname überschreitet die (in der Option HOSTLENGTH) angegebene Länge.
- 30**
Der Pfad überschreitet die (in der Option PATHLENGTH) angegebene Länge.
- 141**
Der Wert für REALMLEN ist nicht positiv oder nicht groß genug für den in der HTTP-Antwort 401 zurückgegebenen Realmwert.
- 13 NOTFND**
RESP2-Werte:
- 155**
Die Informationen zur Anforderungszeile wurden nicht gefunden.
- 19 NOTOPEN**
RESP2-Werte:
- 27**
Das Sitzungstoken ist ungültig.
- 124 TIMEOUT**
RESP2-Werte:
- 62**
Beim Empfangen des Sockets trat eine Zeitlimitüberschreitung auf.

FETCH ANY

Mit dem EXEC CICS-Befehl **FETCH ANY** fragt eine übergeordnete Task den Status aller untergeordneten Tasks ab und gibt die Statuswerte aller abgeschlossenen untergeordneten Tasks zurück, die noch nicht abgerufen wurden.

FETCH ANY



Bedingungen: INVREQ, NOTFND, NOTFINISHED

Dieser Befehl ist threadsicher.

Beschreibung

Mit dem EXEC CICS-Befehl **FETCH ANY** ruft eine übergeordnete Task die Statuswerte aller untergeordneten Tasks ab. Der Befehl **FETCH ANY** gibt eine Antwort zurück, wenn eine infrage kommende Task abge-

geschlossen wurde oder wenn keine untergeordneten Elemente verbleiben, die noch nicht abgerufen wurden.

Optionen

ABCODE(datenbereich)

Gibt den vier Zeichen langen Abbruchcode zurück, wenn die untergeordnete Task abnormal beendet wurde. Wenn kein Abbruch stattgefunden hat, wird die Variable auf Leerzeichen gesetzt.

ANY(datenbereich)

Gibt einen 16 Zeichen umfassenden Binärdatenbereich zurück, in den CICS das abgerufene Token für das untergeordnete Element einfügt. Dieses stimmt mit dem Token überein, das durch den EXEC CICS-Befehl **RUN TRANSID** zurückgegeben wird.

CICS gibt nur das Token einer untergeordneten Task zurück, das noch nicht abgerufen wurde.

CHANNEL(datenbereich)

Gibt den 16 Zeichen langen Namen des Antwortkanals der untergeordneten Task zurück, die durch CHILD angegeben wird. Der Name wird von CICS generiert und ist innerhalb des Geltungsbereichs der aktuellen Verbindungsebene eindeutig. Falls die untergeordnete Task keinen Kanal für die Rückgabe der Antwort besitzt, enthält dieses Feld Leerzeichen.

Ein untergeordneter Kanal kann nur einmal abgerufen werden und der abgerufene Kanal bleibt im Bereich, bis sich die Verbindungsebene ändert. Weitere Informationen zum Geltungsbereich von Kanälen finden Sie unter [Geltungsbereich eines Kanals](#).

COMPSTATUS(cvda)

Gibt einen CVDA-Wert zurück, der den Beendigungsstatus der untergeordneten Task angibt. Die CVDA-Werte lauten wie folgt:

ABEND

Die untergeordnete Task wurde abgebrochen.

NORMAL

Die untergeordnete Task wurde normal beendet.

SECERROR

Die untergeordnete Task konnte aufgrund eines Sicherheitsfehlers nicht angehängt werden. Das Feld ABCODE ist leer.

NOSUSPEND

Gibt an, dass der Befehl sofort eine Rückgabe vornimmt, ohne die Beendigung von gegebenenfalls verbleibenden untergeordneten Tasks abzuwarten. Falls mindestens eines der nicht abgerufenen untergeordneten Elemente abgeschlossen ist, gibt der Befehl eine normale Antwort zurück.

TIMEOUT(datenwert)

Gibt einen Vollwort-Binärdatenwert zurück, der in Millisekunden angibt, wie lange der Befehl maximal auf die Beendigung von verbleibenden untergeordneten Tasks wartet, bevor eine Rückgabe durch den Befehl stattfindet. Der Maximalwert ist 40800000 Millisekunden. Falls die Option TIMEOUT nicht angegeben ist oder der angegebene Wert Null ist, besteht kein Zeitlimit für den Befehl.

Bedingungen

113 NOTFINISHED

RESP2-Werte:

52

Die Option NOSUSPEND war angegeben, aber bei der Ausführung des Befehls waren keine abgeschlossenen untergeordneten Elemente für den Abruf vorhanden.

53

Innerhalb des durch die Option TIMEOUT angegebenen Intervalls wurden keine untergeordneten Elemente abgeschlossen.

13 NOTFND

RESP2-Werte:

1

Es wurden keine untergeordneten Elemente gefunden.

16 INVREQ

RESP2-Werte:

52

Das übergeordnete Element besitzt keine untergeordneten Elemente.

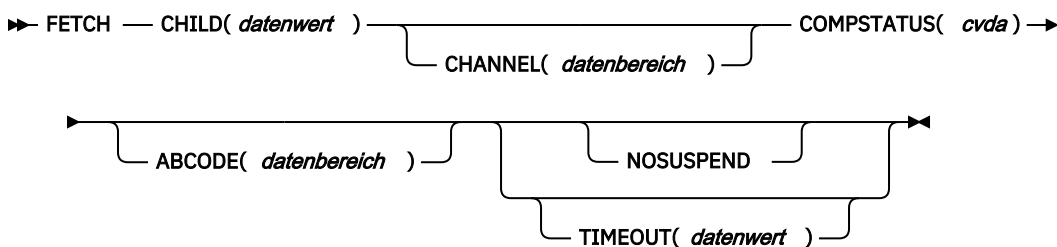
241

Der Wert für TIMEOUT ist nicht gültig.

FETCH CHILD

Der Befehl **EXEC CICS FETCH CHILD** wird von einer übergeordneten Task zum Abfragen des Status einer bestimmten untergeordneten Task verwendet und gibt den Status der angegebenen untergeordneten Task zurück.

FETCH CHILD



Bedingungen: INVREQ, NOTFINISHED

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl **EXEC CICS FETCH CHILD** wird von einer übergeordneten Task zum Abfragen des Status einer bestimmten untergeordneten Task verwendet, die durch das Token CHILD angegeben wird. Der Befehl **FETCH CHILD** führt die Rückgabe aus, wenn die untergeordnete Task abgeschlossen ist.

Optionen

ABCODE(datenbereich)

Gibt den vier Zeichen langen Abbruchcode zurück, wenn die untergeordnete Task abnormal beendet wurde. Wenn kein Abbruch stattgefunden hat, wird die Variable auf Leerzeichen gesetzt.

CHANNEL(datenbereich)

Gibt den 16 Zeichen langen Namen des Antwortkanals der untergeordneten Task zurück, die durch CHILD angegeben wird. Der Name wird von CICS generiert und ist innerhalb des Geltungsbereichs der aktuellen Verbindungsebene eindeutig. Falls die untergeordnete Task keinen Kanal für die Rückgabe der Antwort besitzt, enthält dieses Feld Leerzeichen.

Ein untergeordneter Kanal kann nur einmal abgerufen werden und der abgerufene Kanal bleibt im Bereich, bis sich die Verbindungsebene ändert. Weitere Informationen zum Geltungsbereich von Kanälen finden Sie unter [Geltungsbereich eines Kanals](#).

CHILD(datenwert)

Gibt das untergeordnete Token an, aus dem die Antwort abgerufen werden soll. Das untergeordnete Token wurde durch den Befehl RUN TRANSID zurückgegeben.

COMPSTATUS(cvda)

Gibt einen CVDA-Wert zurück, der den Beendigungsstatus der untergeordneten Task angibt. Die CVDA-Werte lauten wie folgt:

ABEND

Die untergeordnete Task wurde abgebrochen.

NORMAL

Die untergeordnete Task wurde nicht abnormal beendet.

SECERROR

Die untergeordnete Task konnte aufgrund eines Sicherheitsfehlers nicht angehängt werden. Das Feld ABCODE ist leer.

NOSUSPEND

Gibt an, dass der Befehl unverzüglich zurückkehrt, ohne auf den Abschluss der untergeordneten Task zu warten. Wenn die untergeordnete Task bereits abgeschlossen ist, wenn dieser Befehl abgesetzt wird, kehrt der Befehl mit einer normalen Antwort zurück.

TIMEOUT(datenwert)

Gibt einen Vollwort-Binärwert an, der die maximale Zeit in Millisekunden darstellt, die der Befehl auf den Abschluss der angeforderten untergeordneten Task wartet, bevor der Befehl zurückkehrt. Der Maximalwert ist 4080000 Millisekunden. Falls die Option TIMEOUT nicht angegeben ist oder der angegebene Wert Null ist, besteht kein Zeitlimit für den Befehl.

Bedingungen

113 NOTFINISHED

RESP2-Werte:

52

NOSUSPEND wurde angegeben, jedoch ist die untergeordnete Task noch nicht abgeschlossen.

53

Die untergeordnete Task wurde nicht innerhalb des in der Option TIMEOUT angegebenen Intervalls abgeschlossen.

16 INVREQ

RESP2-Werte:

50

Die Daten, die in der Option CHILD angegeben wurden, sind kein gültiges Token für die untergeordnete Task oder die untergeordnete Task wurde bereits durch einen Befehl **FREE CHILD** freigegeben.

51

Der Kanal der untergeordneten Task wurde bereits abgerufen.

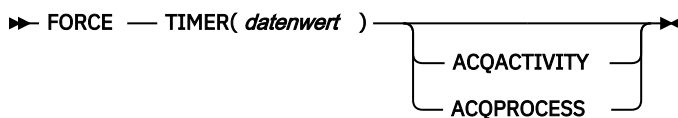
241

Der Wert für TIMEOUT ist nicht gültig.

FORCE TIMER

Erzwingt den vorzeitigen Ablauf eines BTS-Zeitgebers.

FORCE TIMER



Bedingungen: INVREQ, TIMERERR

Beschreibung

Der Befehl FORCE TIMER zwingt einen BTS-Zeitgeber, der noch nicht abgelaufen ist, zum sofortigen Ablauf. Dadurch wird das Ereignis, das dem Zeitgeber zugeordnet ist, ausgelöst.

Wenn der Zeitgeber bereits abgelaufen ist, hat der Befehl keine Wirkung.

Die Aktivität, die Eigner des Zeitgebers ist, kann wie folgt angegeben werden:

- Explizit durch Angabe der Option ACQPROCESS oder ACQACTIVITY.
- Implizit durch Weglassen der Optionen ACQPROCESS und ACQACTIVITY. Wenn diese Optionen nicht angegeben werden, wird die aktuelle Aktivität impliziert.

Optionen

ACQACTIVITY

Gibt eine der folgenden Informationen an:

- Wenn das Programm, das den Befehl absetzt, einen Prozess angefordert hat, gibt diese Option an, dass der Eigner des Zeitgebers die Stammaktivität dieses Prozesses ist.
- Andernfalls gibt diese Option an, dass der Zeitgeber der Aktivität gehört, die von dem Programm durch einen Befehl ACQUIRE ACTIVITYID angefordert wurde.

ACQPROCESS

Gibt an, dass der Zeitgeber dem Prozess gehört, den das Programm, das den Befehl absetzt, in der aktuellen Arbeitseinheit angefordert hat.

TIMER(datenwert)

Gibt den Namen (1 - 16 Zeichen) des Zeitgebers an, dessen Ablauf erzwungen werden soll.

Bedingungen

16 INVREQ

RESP2-Werte:

1

Der Befehl wurde außerhalb des Gültigkeitsbereichs einer momentan aktiven Aktivität abgesetzt.

16

Die Option ACQPROCESS wurde angegeben, jedoch ist kein angeforderter Prozess vorhanden.

17

Die Option ACQACTIVITY wurde angegeben, jedoch ist keine angeforderte Aktivität vorhanden.

115 TIMERERR

RESP2-Werte:

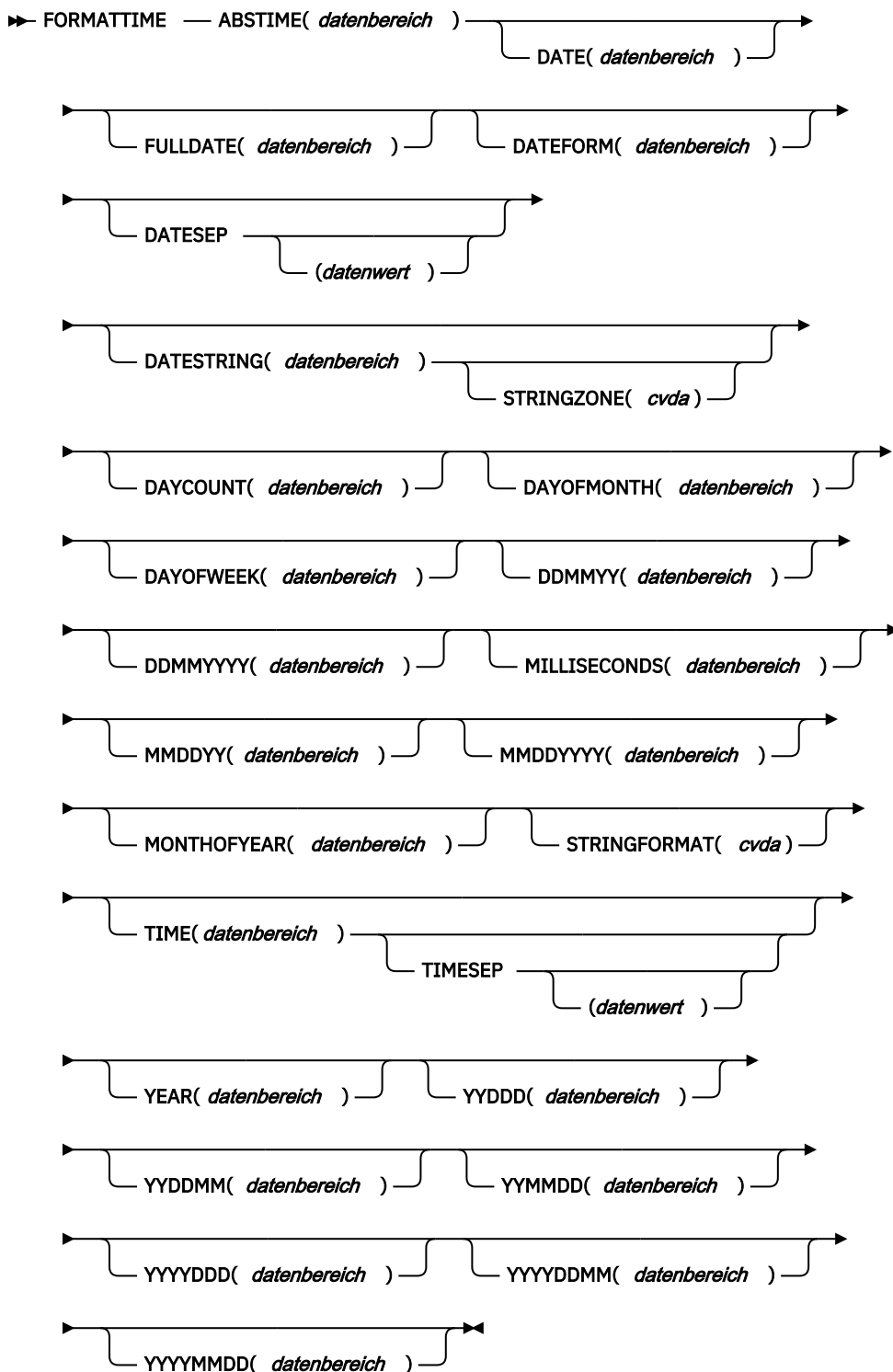
13

Der Zeitgeber, der in der Option TIMER angegeben wurde, ist nicht vorhanden.

FORMATTIME

Wandelt das absolute Datum und die absolute Uhrzeit in ein angegebenes Format um.

FORMATTIME



Bedingung: INVREQ

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl `FORMATTIME` wandelt das absolute Datum und die absolute Uhrzeit in eines von verschiedenen Formaten um. Normalerweise ist das Argument `ABSTIME` der Wert, der durch einen Befehl `ASKTIME` `ABSTIME` zurückgegeben wird.

Um die abgelaufene Zeit in einem bestimmten Format abzurufen, kann der Datenwert von `ABSTIME` der Unterschied zwischen zwei Werten sein, die durch `ASKTIME` zurückgegeben werden, und Optionen wie `DAYCOUNT(d)` und `TIME(t)` können angegeben werden.

Wenn Sie mit der Option `DATESTRING` eine konzipierte Datums- und Zeitmarkenzeichenfolge anfordern, für die die Systemzeit im UTC-Format vorliegen muss, berechnet CICS den erforderlichen Zeitonenabstand aus dem von Ihnen angegebenen Wert für `ABSTIME`, der in Ortszeit vorliegt, und generiert die Datums- und Zeitmarkenzeichenfolge in UTC-Zeit. Alle anderen Werte, die durch den Befehl `FORMATTIME` zurückgegeben werden, wie zum Beispiel der Wert für `TIME`, werden in Ortszeit zurückgegeben. Daher ist es für den Befehl `FORMATTIME` normal, eine konzipierte Datums- und Zeitmarkenzeichenfolge zu generieren, die ein Datum und eine Uhrzeit zeigt, und andere Werte zurückzugeben, die für den gleichen angegebenen Wert in `ABSTIME` ein anderes Datum und eine andere Uhrzeit zeigen.

Optionen

ABSTIME(datenbereich)

Gibt den Datenbereich für die Anzahl Millisekunden seit dem 1. Januar 1900 (00:00 Uhr) an, der als absolute Zeit bezeichnet wird. Die Zeit wird der Uhrzeit der Systemuhr mit Anpassung im Hinblick auf Schaltsekunden und Ortszeitunterschiede (einschließlich Sommerzeit) entnommen, auf die Millisekunde abgeschnitten und als gepacktes Dezimalformat mit einer Länge von 8 Byte zurückgegeben. Mit dem Befehl `FORMATTIME` können Sie die Daten in andere gängige Formate ändern.

Der Parameter hat das folgende Format:

```
COBOL: PIC S9(15) COMP-3  
C:      char data_ref[8];  
PL/I:   FIXED DEC(15);  
ASM:    PL8
```

DATE(datenbereich)

Gibt die Variable an, die das Datum in dem Format empfangen soll, das im Systeminitialisierungsparameter `DATFORM` angegeben ist. Trennzeichen sind vorhanden, wenn sie durch die Option `DATESEP` angefordert werden. Wird kein Trennzeichen angefordert, ist der zurückgegebene Wert in dem acht Zeichen langen Benutzerfeld linksbündig ausgerichtet. Sie verwenden diese Option normalerweise nur, wenn ein Datum zu Ausgabezwecken benötigt wird. Wenn ein Datum zur Analyse benötigt wird, fordern Sie das Datum in expliziter Form, zum Beispiel mit der Option `MMDDYY`, an.

DATEFORM(datenbereich)

Gibt das Format des installationsdefinierten Datums an. CICS gibt `YYMMDD`, `DDMMYY` oder `MMDDYY` (sechs Zeichen) entsprechend dem Systeminitialisierungsparameter `DATFORM` zurück.

DATESEP(datenwert)

Gibt das Zeichen an, das als Trennzeichen zwischen Jahr und Monat und zwischen Tag und Monat bzw. bei Angabe des Formats `YYDDD` zwischen Jahr und Tag, eingefügt werden soll.

Wenn Sie diese Option nicht angeben, wird kein Trennzeichen bereitgestellt. Wenn Sie *datenwert* nicht angeben, wird ein Schrägstrich (/) als Trennzeichen angenommen.

DATESTRING(datenbereich)

Gibt ein 64 Zeichen langes Benutzerfeld an, in dem CICS die konzipierte Datums- und Zeitmarkenzeichenfolge in dem Format zurückgibt, das in der Option `STRINGFORMAT` angegeben wird. Wenn `STRINGFORMAT` nicht angegeben wird, wird als Standardformat das RFC 1123-Format (RFC1123) bereitgestellt. Wenn Sie die Option `DATESTRING` verwenden, führen Sie zunächst den Befehl `ASKTIME` `ABSTIME` aus, um einen Wert für die Option `ABSTIME` abzurufen. Wenn der Wert für die Option

ABSTIME aus einer anderen Quelle stammt, ist die konzipierte Datums- und Zeitmarkenzeichenfolge, die durch den Befehl FORMATTIME zurückgegeben wird, möglicherweise nicht korrekt.

DAYCOUNT(*datenbereich*)

Gibt die Anzahl von Tagen seit dem 1. Januar 1900 (Tag 1) als Vollwort-Binärwert zurück. Diese Funktion ist nützlich, wenn Sie das aktuelle Datum mit einem früheren Datum vergleichen müssen, das zum Beispiel in einer Datei (Dataset) gespeichert wurde.

DAYOFMONTH(*datenbereich*)

Gibt die Zahl des Tages im Monat als Vollwort-Binärzahl zurück.

DAYOFWEEK(*datenbereich*)

Gibt die relative Zahl des Tages in der Woche als Vollwort-Binärzahl zurück: Sonntag=0, Samstag=6. Diese Zahl kann in eine Textform für den Tag in einer beliebigen Sprache konvertiert werden.

DDMMYY(*datenbereich*)

Gibt das acht Zeichen lange Benutzerfeld an, in dem CICS das Datum im Format Tag/Monat/Jahr zurückgibt. Beispiel: 21/10/98. Trennzeichen sind vorhanden, wenn sie durch die Option DATESEP angefordert werden. Wird kein Trennzeichen angefordert, ist der zurückgegebene Wert in dem acht Zeichen langen Benutzerfeld linksbündig ausgerichtet.

DDMMYYYY(*datenbereich*)

Gibt das zehn Zeichen lange Benutzerfeld an, in dem CICS das Datum im Format Tag/Monat/Jahr zurückgibt. Beispiel: 17/06/1995. Trennzeichen sind vorhanden, wenn sie durch die Option DATESEP angefordert werden. Wird kein Trennzeichen angefordert, ist der zurückgegebene Wert in dem zehn Zeichen langen Benutzerfeld linksbündig ausgerichtet.

FULLDATE(*datenbereich*)

Gibt das zehn Zeichen lange Benutzerfeld an, in dem CICS das Datum in dem Format zurückgibt, das im Systeminitialisierungsparameter DATFORM angegeben ist, wobei das Jahr vierstellig angegeben wird. Trennzeichen sind vorhanden, wenn sie durch die Option DATESEP angefordert werden. Wird kein Trennzeichen angefordert, ist der zurückgegebene Wert in dem zehn Zeichen langen Benutzerfeld linksbündig ausgerichtet. Sie verwenden diese Option normalerweise nur, wenn ein Datum zu Ausgabezwecken benötigt wird. Wenn ein Datum zur Analyse benötigt wird, fordern Sie das Datum in expliziter Form, zum Beispiel mit der Option MMDDYYYY, an.

MILLISECONDS(*datenbereich*)

Gibt die Anzahl Millisekunden in der aktuellen Sekunde, die durch ABSTIME angegeben wird, als binäre Ganzzahl aus dem Bereich 0 - 999 zurück.

MMDDYY(*datenbereich*)

Gibt das acht Zeichen lange Benutzerfeld an, in dem CICS das Datum im Format Monat/Tag/Jahr zurückgibt. Beispiel: 10/21/95. Trennzeichen sind vorhanden, wenn sie durch die Option DATESEP angefordert werden. Wird kein Trennzeichen angefordert, ist der zurückgegebene Wert in dem acht Zeichen langen Benutzerfeld linksbündig ausgerichtet.

MMDDYYYY(*datenbereich*)

Gibt das zehn Zeichen lange Benutzerfeld an, in dem CICS das Datum im Format Monat/Tag/Jahr zurückgibt. Beispiel: 11/21/1995. Trennzeichen sind vorhanden, wenn sie durch die Option DATESEP angefordert werden. Wird kein Trennzeichen angefordert, ist der zurückgegebene Wert in dem zehn Zeichen langen Benutzerfeld linksbündig ausgerichtet.

MONTHOFYEAR(*datenbereich*)

Der Wert *datenbereich* wird auf die relative Monatszahl des Jahres als Vollwort-Binärzahl gesetzt (Januar=1, Dezember=12). Sie können diese Zahl in Ihrem Anwendungsprogramm in den Namen des Monats einer beliebigen Sprache konvertieren.

STRINGFORMAT(*cvda*)

Gibt das Format für die konzipierte Datums- und Zeitmarkenzeichenfolge an, die in DATESTRING zurückgegeben wird. Die folgenden CVDA-Werte sind möglich:

RFC1123

Gibt das RFC 1123-Format an, das für die Verwendung in HTTP-Nachrichten geeignet ist. Diese Datums- und Zeitmarkenzeichenfolge enthält den Tag, das Datum und die Uhrzeit im 24-Stunden-

Format. Beispiel: "Tue, 01 Apr 2003 10:01:02 +0000". Dieses Format enthält keine Millisekunden und die Anzahl der Sekunden wird abgeschnitten.

RFC3339

Gibt das RFC 3339-Format an, das auch als XML-Datentyp 'dateTime' bekannt ist. Dieses Format ist eine Implementierung eines Teils des ISO 8601-Standards. Ein Beispiel für eine Datums- und Zeitmarke in diesem Format ist 2003-04-24T10:01:02+00:00. Datums- und Zeitmarken in diesem Format sind in UTC (Coordinated Universal Time, koordinierte Weltzeit). Diese Datums- und Zeitmarkenzeichenfolge enthält das Datum und die Uhrzeit im 24-Stunden-Format. Der Zeitzoneneunterschied (-12:00 bis +12:00) wird am Ende der Datums- und Zeitmarke angegeben.

Die RFC 3339-Spezifikation lässt die Verwendung des Buchstabens Z für keinen Unterschied (+00:00) zu. Ein Dezimalbruch einer Sekunde im 24-Stunden-Format ist in der Spezifikation optional und der Befehl FORMATTIME enthält diesen nicht. Ein Beispiel für eine Zeitmarke, die den Dezimalbruch einer Sekunde und den Buchstaben Z für keinen Zeitonenunterschied enthält, ist 2003-04-01T10:01:02.498Z. Wenn Sie den Dezimalbruch einer Sekunde in Ihrer Anwendung hinzufügen wollen, können Sie die Option MILLISECONDS verwenden, um die Anzahl der Millisekunden zurückzugeben, die ebenfalls abgelaufen sind.

STRINGZONE(*cvda*)

Gibt die Zeitzone an, in der die Zeitmarke, die in DATESTRING zurückgegeben wird, zurückgegeben werden soll. Die CVDA-Werte lauten wie folgt:

UTC

DATESTRING soll in UTC zurückgegeben werden. Dies ist die Standardeinstellung.

LOCAL

DATESTRING soll in der Ortszeitzone zurückgegeben werden.

TIME(*datenbereich*)

Der Wert *datenbereich* wird als acht Zeichen langes Feld auf die aktuelle Uhrzeit im 24-Stunden-Format in der Form hh:mm:ss gesetzt, wobei das Trennzeichen durch die Option TIMESEP angegeben wird. Die Anzahl der Sekunden wird abgeschnitten. Verwenden Sie die Option MILLISECONDS, um die Anzahl der Millisekunden zurückzugeben, die ebenfalls abgelaufen sind.

TIMESEP(*datenwert*)

Gibt das Zeichen an, das als Trennzeichen in der zurückgegebenen Zeit verwendet werden soll. Wenn Sie diese Option nicht angeben, wird kein Trennzeichen angenommen und es werden 6 Byte in einem 8 Zeichen langen Feld zurückgegeben. Wenn Sie *datenwert* nicht angeben, wird ein Doppelpunkt (:) als Trennzeichen verwendet.

YEAR(*datenbereich*)

Gibt eine volle vierstellige Zahl für das Jahr als Vollwort-Binärzahl zurück. Beispiele: 1995, 2001.

YYDDD(*datenbereich*)

Gibt das sechs Zeichen lange Benutzerfeld an, in dem CICS das Datum im Format Jahr/Tag zurückgibt. Beispiel: 95/301. Trennzeichen sind vorhanden, wenn sie durch die Option DATESEP angefordert werden. Wird kein Trennzeichen angefordert, ist der zurückgegebene Wert in dem sechs Zeichen langen Benutzerfeld linksbündig ausgerichtet.

YYDDMM(*datenbereich*)

Gibt das acht Zeichen lange Benutzerfeld an, in dem CICS das Datum im Format Jahr/Tag/Monat zurückgibt. Beispiel: 95/30/10. Trennzeichen sind vorhanden, wenn sie durch die Option DATESEP angefordert werden. Wird kein Trennzeichen angefordert, ist der zurückgegebene Wert in dem acht Zeichen langen Benutzerfeld linksbündig ausgerichtet.

YYMMDD(*datenbereich*)

Gibt das acht Zeichen lange Benutzerfeld an, in dem CICS das Datum im Format Jahr/Monat/Tag zurückgibt. Beispiel: 95/10/21. Trennzeichen sind vorhanden, wenn sie durch die Option DATESEP angefordert werden. Wird kein Trennzeichen angefordert, ist der zurückgegebene Wert in dem acht Zeichen langen Benutzerfeld linksbündig ausgerichtet.

YYYYDDD(*datenbereich*)

Gibt das acht Zeichen lange Benutzerfeld an, in dem CICS das Datum im Format Jahr/Tag zurückgibt. Beispiel: 1995/200. Trennzeichen sind vorhanden, wenn sie durch die Option DATESEP angefordert

werden. Wird kein Trennzeichen angefordert, ist der zurückgegebene Wert in dem acht Zeichen langen Benutzerfeld linksbündig ausgerichtet.

YYYYDDMM(datenbereich)

Gibt das zehn Zeichen lange Benutzerfeld an, in dem CICS das Datum im Format Jahr/Tag/Monat zurückgibt. Beispiel: 1995/21/06. Trennzeichen sind vorhanden, wenn sie durch die Option DATESEP angefordert werden. Wird kein Trennzeichen angefordert, ist der zurückgegebene Wert in dem zehn Zeichen langen Benutzerfeld linksbündig ausgerichtet.

YYYYMMDD(datenbereich)

Gibt das zehn Zeichen lange Benutzerfeld an, in dem CICS das Datum im Format Jahr/Monat/Tag zurückgibt. Beispiel: 1995/06/21. Trennzeichen sind vorhanden, wenn sie durch die Option DATESEP angefordert werden. Wird kein Trennzeichen angefordert, ist der zurückgegebene Wert in dem zehn Zeichen langen Benutzerfeld linksbündig ausgerichtet.

Bedingungen

16 INVREQ

RESP2-Werte:

1

Der Wert von ABSTIME ist kleiner als null oder nicht im gepackten Dezimalzahlformat.

2

Ungültiger CVDA-Wert für die Option STRINGFORMAT.

Standardaktion: Die Task wird abnormal beendet.

Beispiele

Das folgende Beispiel zeigt den Effekt einiger der Optionen des Befehls. Dabei soll *utime* den Wert 003578979940458 in Millisekunden enthalten.

```
EXEC CICS ASKTIME ABSTIME(utime)
EXEC CICS FORMATTIME ABSTIME(utime)
          DATESEP('-') DDMMYY(date)
          TIME(time) TIMESEP(':')
```

In diesem Beispiel wird 05-31-13 für *date* und 08:05:40 für *time* zurückgegeben.

FREE

Gibt ein Terminal oder eine logische Einheit zurück.

FREE

➡ FREE →

Bedingung: NOTALLOC

Beschreibung

Der Befehl FREE gibt ein Terminal oder eine logische Einheit zurück, wenn es bzw. sie von der Transaktion, der das Terminal bzw. die logische Einheit gehört, nicht mehr benötigt wird. Die Hauptfunktion wird freigegeben.

Wenn Sie Execution Diagnostic Facility (EDF) ausführen und die Transaktion die Hauptfunktion freigibt, wird EDF beendet.

Bedingungen

61 NOTALLOC

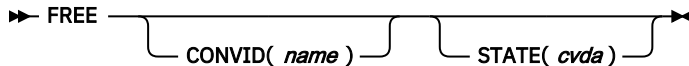
Tritt auf, wenn die Task dem Terminal nicht zugeordnet ist.

Standardaktion: Die Task wird abnormal beendet.

FREE (APPC)

Gibt eine zugeordnete APPC-Sitzung an CICS zurück.

FREE (APPC)



Bedingungen: INVREQ, NOTALLOC

Beschreibung

Der Befehl gibt eine APPC-Sitzung an CICS zurück, wenn eine Transaktion, der sie gehört, sie nicht länger benötigt. Die Sitzung kann anschließend zur Verwendung durch andere Transaktionen zugeordnet werden.

Wenn Sie die Option CONVID nicht angeben, wird die Hauptfunktion freigegeben. Funktionen, die nicht explizit freigegeben werden, werden von CICS bei Beendigung der Task freigegeben.

Wenn Sie Execution Diagnostic Facility (EDF) ausführen und die Transaktion die Hauptfunktion freigibt, wird EDF beendet.

Optionen

CONVID(name)

Gibt die zugeordnete APPC-Sitzung an, die freigegeben werden soll. Der vier Zeichen lange Name bezeichnet entweder das Token, das von einem zuvor ausgeführten Befehl ALLOCATE in EIBRSRCE im EIB zurückgegeben wurde, oder das Token, das die Hauptsitzung (die durch einen zuvor ausgeführten Befehl ASSIGN zurückgegeben wurde) darstellt.

Wenn diese Option nicht angegeben wird, wird die Hauptfunktion angenommen.

STATE(cvda)

Ruft den Status des aktuellen Datenaustausches ab. Die Option STATE für einen Befehl FREE gibt den CVDA-Code 00 zurück, wenn kein aktiver Dialog mehr vorhanden ist. Die weiteren CVDA-Ausgabewerte sind folgende:

- ALLOCATED
- CONFFREE
- CONFRECEIVE
- CONFSEND
- FREE
- PENDFREE
- PENDRECEIVE
- RECEIVE
- ROLLBACK
- SEND

- SYNCFREE
- SYNCRECEIVE
- SYNCSEND

Bedingungen

16 INVREQ

RESP2-Werte:

200

Eine Serveranwendung für Distributed Program Link hat in der Option CONVID die Sitzung für die Funktionsverlagerung (ihre Hauptfunktion) angegeben.

Tritt außerdem in jedem der folgenden Fälle auf (RESP2 nicht festgelegt):

- Der Wert von CONVID, der in dem Befehl angegeben wurde, bezieht sich auf einen APPC-Basisdialog (nicht auf einen formatfreien APPC-Dialog).
- Der Wert von CONVID, der in dem Befehl angegeben wurde, bezieht sich auf einen CPI Communications-Dialog.

Standardaktion: Die Task wird abnormal beendet.

61 NOTALLOC

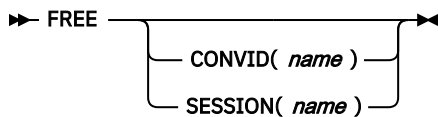
Tritt auf, wenn sich der in CONVID angegebene Wert nicht auf einen Dialog bezieht, dessen Eigner die Anwendung ist.

Standardaktion: Die Task wird abnormal beendet.

FREE (LUTYPE6.1)

Gibt LUTYPE6.1-Sitzungen an CICS zurück.

FREE (LUTYPE6.1)



Bedingungen: INVREQ, NOTALLOC

Beschreibung

Der Befehl gibt eine LUTYPE6.1-Sitzung an CICS zurück, wenn eine Transaktion, der sie gehört, sie nicht länger benötigt. Die Sitzung kann anschließend zur Verwendung durch andere Transaktionen zugeordnet werden.

Wenn Sie weder CONVID noch SESSION angeben, wird die Hauptfunktion freigegeben. Funktionen, die nicht explizit freigegeben werden, werden von CICS bei Beendigung der Task freigegeben.

Wenn Sie Execution Diagnostic Facility (EDF) ausführen und die Transaktion die Hauptfunktion freigibt, wird EDF beendet.

Optionen

CONVID(name)

Gibt die LUTYPE6.1-Sitzung an, die freigegeben werden soll. Der vier Zeichen lange Name bezeichnet entweder das Token, das von einem zuvor ausgeführten Befehl ALLOCATE in EIBRSRCE im EIB zurückgegeben wurde, oder das Token, das die Hauptsitzung (die durch einen zuvor ausgeführten Befehl ASSIGN zurückgegeben wurde) darstellt.

SESSION(name)

Gibt die symbolische ID (1-4 Zeichen) für einen Terminaleintrag in der Terminalsteuertabelle (TCTTE) für die Sitzung an. Diese Option gibt die alternative Funktion an, die verwendet werden soll.

Bedingungen**16 INVREQ**

Tritt auf, wenn die im Befehl angegebene Sitzung für einen (nicht formatfreien) APPC-Basisdialog zugeordnet wurde.

Siehe auch EIBRCODE in [EIB-Felder](#).

Standardaktion: Die Task wird abnormal beendet.

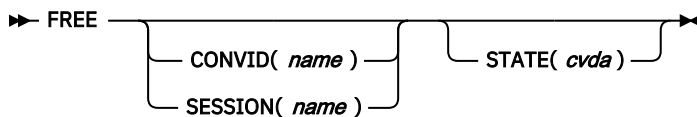
61 NOTALLOC

Tritt auf, wenn die im Befehl angegebene Sitzung nicht der Anwendung gehört.

Standardaktion: Die Task wird abnormal beendet.

FREE (MRO)

Gibt MRO-Sitzungen an CICS zurück.

FREE (MRO)

Bedingungen: INVREQ, NOTALLOC

Beschreibung

Der Befehl gibt eine MRO-Sitzung an CICS zurück, wenn eine Transaktion, der sie gehört, sie nicht länger benötigt. Die Sitzung kann anschließend zur Verwendung durch andere Transaktionen zugeordnet werden.

Wenn Sie weder CONVID noch SESSION angeben, wird die Hauptfunktion freigegeben. Funktionen, die nicht explizit freigegeben werden, werden von CICS bei Beendigung der Task freigegeben.

Wenn Sie Execution Diagnostic Facility (EDF) ausführen und die Transaktion die Hauptfunktion freigibt, wird EDF beendet.

Optionen**CONVID(name)**

Gibt die MRO-Sitzung an, die freigegeben werden soll. Der vier Zeichen lange Name bezeichnet entweder das Token, das von einem zuvor ausgeführten Befehl ALLOCATE in EIBRSRCE im EIB zurückgegeben wurde, oder das Token, das die Hauptsitzung (die durch einen zuvor ausgeführten Befehl ASSIGN zurückgegeben wurde) darstellt.

SESSION(name)

Gibt die symbolische ID (1-4 Zeichen) für einen Terminaleintrag in der Terminalsteuertabelle (TCTTE) für die Sitzung an. Diese Option gibt die zu verwendende alternative Funktion an.

STATE(cvda)

Ruft den Status des aktuellen Dialogs ab. Die Option STATE für einen Befehl FREE gibt den CVDA-Code 00 zurück, wenn kein aktiver Dialog mehr vorhanden ist. Die weiteren CVDA-Ausgabewerte sind folgende:

- ALLOCATED
- FREE

- PENDFREE
- RECEIVE
- ROLLBACK
- SEND
- SYNCFREE
- SYNCRECEIVE
- SYNCSEND

Bedingungen

16 INVREQ

Tritt in einer der folgenden Situationen auf:

- Die im Befehl angegebene Sitzung wurde für einen (nicht formatfreien) APPC-Basisdialog zugeordnet.
- Die Sitzung ist im falschen Status, um freigegeben zu werden.

Siehe auch EIBRCODE in [EIB-Felder](#).

Standardaktion: Die Task wird abnormal beendet.

61 NOTALLOC

Tritt auf, wenn die im Befehl angegebene Sitzung nicht der Anwendung gehört.

Standardaktion: Die Task wird abnormal beendet.

FREE CHILD

Mit dem EXEC CICS-Befehl **FREE CHILD** wird ein angegebenes untergeordnetes Token freigegeben, das zuvor durch einen EXEC CICS-Befehl **RUN TRANSID** zugeordnet wurde. Falls eine übergeordnete Task die Antwort einer untergeordneten Task nicht mehr benötigt, gibt der Befehl **FREE CHILD** die Ressourcen, die dieser untergeordneten Task zugeordnet sind, bei ihrem Abschluss frei, statt auf ihren Abruf zu warten.

FREE CHILD

➡ FREE — CHILD(*datenwert*) ➡

Bedingung: INVREQ

Dieser Befehl ist threadsicher.

Beschreibung

Mit dem Befehl **FREE CHILD** gibt eine übergeordnete Task ein untergeordnetes Token frei, das durch einen vorherigen Befehl **RUN TRANSID** zugeordnet wurde. Falls die zugehörige untergeordnete Task bereits abgeschlossen wurde, wird das dem Token zugeordnete Speicherobjekt sofort freigegeben; außerdem gibt CICS jeden untergeordneten Kanal frei (unter der Voraussetzung, dass er nicht mit einem EXEC CICS-Befehl **FETCH** abgerufen wurde).

Falls die untergeordnete Task nicht abgeschlossen ist, wird das Speicherobjekt zusammen mit jedem untergeordneten Kanal beim Abschluss der untergeordneten Task freigegeben. Nachdem dieser Befehl ausgegeben wurde, schlagen alle nachfolgenden Versuche, das Token mit dem Befehl **FETCH CHILD** zu referenzieren, mit der Bedingung INVREQ fehl; des Weiteren wird das Token nicht durch gegebenenfalls nachfolgende Befehle **FETCH ANY** zurückgegeben.

CICS gibt alle durch den Befehl **RUN TRANSID** erstellten untergeordneten Tokens implizit frei, sobald die übergeordnete Task beendet wird.

Optionen

CHILD(*datenwert*)

Gibt das untergeordnete Token an, das freigegeben werden soll. Das untergeordnete Token wurde durch den Befehl **RUN TRANSID** zurückgegeben.

Bedingungen

16 INVREQ

RESP2-Werte:

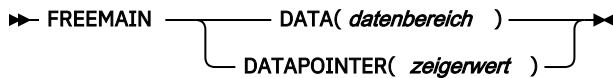
50

Bei den in der Option CHILD angegebenen Daten handelt es sich nicht um ein gültiges untergeordnetes Token oder die untergeordnete Task wurde bereits durch einen Befehl **FREE CHILD** freigegeben.

FREEMAIN

Gibt den Hauptspeicher frei, der durch eine GETMAIN-Anforderung angefordert wurde.

FREEMAIN



Bedingung:

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl FREEMAIN gibt den folgenden Speicher frei:

- Den Hauptspeicher, der von einem Befehl GETMAIN angefordert wurde, der durch die Anwendung abgesetzt wurde.
- Den Hauptspeicher, der durch einen Befehl LOAD für ein Programm, eine Maske oder eine Tabelle angefordert wurde, das bzw. die mit RELOAD=YES definiert ist.

Wenn die Task, die den Speicher angefordert oder das Programm geladen hat, den Speicher nicht freigibt, gibt CICS den Speicher bei Beendigung der Task frei, mit Ausnahme der folgenden Situationen:

- Der Befehl GETMAIN wurde mit der Option SHARED angegeben. Der Speicher bleibt zugeordnet, bis eine andere Task eine Anforderung FREEMAIN oder FREEMAIN64 zur Freigabe absetzt.
- Das Programm ist mit der Einstellung RELOAD=YES definiert. Der Speicher bleibt zugeordnet, bis eine andere Task eine Anforderung FREEMAIN oder FREEMAIN64 zur Freigabe absetzt.
- Das Programm ist mit RELOAD=NO definiert, jedoch wurde es mit der Option HOLD geladen. Das Programm bleibt verfügbar, bis es durch eine andere Task freigegeben wird.

Anmerkung: In den ersten beiden aufgeführten Situationen kann die Verwendung von FREEMAIN möglicherweise transaktionsübergreifende Affinitäten erstellen, die sich negativ auf die Verwendung des dynamischen Transaktionsroutings auswirken. Weitere Informationen zu Transaktionsaffinitäten finden Sie unter [Affinität](#).

Sie können CICS-Schlüsselspeicher aus einem Programm nur freigeben, wenn es im CICS-Schlüssel ausgeführt wird. Wenn der zuvor angeforderte Speicher aus CICS-Schlüsselspeicher abgerufen wurde und sich das Programm, das die Anforderung FREEMAIN absetzt, im Benutzerschlüssel befindet, tritt eine Bedingung INVREQ mit dem RESP2-Wert 2 auf.

Zur Freigabe von Hauptspeicher, der durch eine Anforderung GETMAIN64 abgerufen wurde, verwenden Sie den Befehl FREEMAIN64. Weitere Informationen finden Sie unter [FREEMAIN64](#).

Optionen

DATA(*datenbereich*)

Gibt den Datenbereich des Hauptspeichers an, der freigegeben werden soll.

In Assemblersprache muss *datenbereich* ein verschiebbarer Ausdruck sein, der eine Datenreferenz ist. In COBOL oder C muss er ein Datennamen sein. In PL/I muss er eine Datenreferenz sein.

Die Länge des freigegebenen Speichers ist die Länge, die durch die Anforderung GETMAIN abgerufen wurde und nicht unbedingt die Länge des Datenbereichs.

DATAPointer(*zeigerwert*)

Gibt die Adresse des Hauptspeichers, der freigegeben werden soll, als Zeigerverweis an. Diese Option ist eine Alternative zur Option DATA und gibt den Zeigerverweis an, der durch einen Befehl GETMAIN mit der Option SET zurückgegeben wurde.

Die Länge des freigegebenen Speichers ist die Länge, die durch die Anforderung GETMAIN abgerufen wurde.

Bedingungen

16 INVREQ

RESP2-Werte:

1

Der Speicher, der durch den Parameter DATA oder DATAPointer angegeben wurde, ist kein Speicher, der durch einen Befehl GETMAIN abgerufen wurde.

2

Der Speicherbereich, der durch den Parameter DATA oder DATAPointer angegeben wurde, befindet sich im CICS-Schlüsselspeicher und das Programm, das den Befehl FREEMAIN abgesetzt hat, befindet sich im Benutzerschlüssel.

3

Der Speicherbereich, der durch den Parameter DATA oder DATAPointer angegeben wurde, wird von CICS verwaltet und jede Anforderung FREEMAIN, die versucht, CICS-verwalteten Speicherbereiche freizugeben, wird zurückgewiesen.

Standardaktion: Die Task wird abnormal beendet.

Beispiel: COBOL

```
DATA DIVISION.
WORKING-STORAGE SECTION.
77 AREA-POINTER    USAGE IS POINTER.
LINKAGE SECTION.
01 WORKAREA        PIC X(100).
PROCEDURE DIVISION.
EXEC CICS GETMAIN SET(AREA-POINTER)
LENGTH(100)
END-EXEC.

SET ADDRESS OF WORKAREA TO AREA-POINTER.
.
EXEC CICS FREEMAIN DATA(WORKAREA)
END-EXEC.
EXEC CICS RETURN
END-EXEC.
```

Alternativ könnte im vorherigen COBOL-Beispiel der Speicher mit dem folgenden Befehl freigegeben werden:

```
EXEC CICS FREEMAIN DATAPointer(AREA-POINTER)
END-EXEC.
```

Beispiel: C

```
#pragma XOPTS(CICS);
#define MAINSIZE 100;
main()
{
    char          *buffer;
    struct eib_record dfheiptr;
    EXEC CICS ADDRESS EIB(dfheiptr);
    EXEC CICS GETMAIN SET(buffer)
                      LENGTH(MAINSIZE);
    buffer[2] = 'a';
    .
    .
    EXEC CICS FREEMAIN DATA(buffer);
    EXEC CICS RETURN;
}
```

Beispiel: PL/I

```
DCL AREA_PTR    POINTER,
      WORKAREA   CHAR(100) BASED(AREA_PTR);
.
.
EXEC CICS GETMAIN SET(AREA_PTR) LENGTH(100);
.
EXEC CICS FREEMAIN DATA(WORKAREA);
```

Beispiel: Assembler

```
WORKAREA    DS    CL100
.
.
                EXEC CICS GETMAIN SET(9) LENGTH(100)
                USING  WORKAREA,9
                EXEC CICS FREEMAIN DATA(WORKAREA)
```

Alternativ können Sie den Speicher mit der Option DATAPOINTER wie im folgenden Beispiel angeben:

```
WORKAREA    DS    CL100
.
                EXEC CICS GETMAIN SET(9) LENGTH(100)
                USING  WORKAREA,9
.
.
                DROP    9
.
                EXEC CICS FREEMAIN DATAPOINTER(9)
```

FREEMAIN64

Gibt Speicher frei, der durch eine GETMAIN- oder GETMAIN64-Anforderung angefordert wurde. Dieser Befehl ist nur zur Verwendung in AMODE(64)-Anwendungsprogrammen in Assemblersprache vorgesehen, die nicht über Language Environment (LE) ausgeführt werden.

Weitere Informationen finden Sie unter [Beschränkungen und Anforderungen an die Assemblersprachprogrammierung](#).

FREEMAIN64

➡ FREEMAIN64 DATA(*datenbereich64*)
DATAPOINTER(*zeigerwert64*)

Bedingung: INVREQ

Dieser Befehl ist threadsicher.

Beschreibung

FREEMAIN64 gibt den folgenden Speicher frei:

- Den Hauptspeicher, der von einem Befehl GETMAIN oder GETMAIN64 angefordert wurde, der durch die Anwendung abgesetzt wurde.
- Den Hauptspeicher, der durch einen Befehl LOAD für ein Programm, eine Maske oder eine Tabelle angefordert wurde, das bzw. die mit RELOAD=YES definiert ist.

Wenn die Task, die den Speicher angefordert oder das Programm geladen hat, den Speicher nicht freigibt, gibt CICS den Speicher bei Beendigung der Task frei, mit Ausnahme der folgenden Situationen:

- Der Befehl GETMAIN oder GETMAIN64 wurde mit der Option SHARED angegeben. Der Speicher bleibt zugeordnet, bis eine andere Task eine Anforderung FREEMAIN oder FREEMAIN64 zur Freigabe absetzt.
- Das Programm ist mit der Einstellung RELOAD=YES definiert. Der Speicher bleibt zugeordnet, bis eine andere Task eine Anforderung FREEMAIN oder FREEMAIN64 zur Freigabe absetzt.
- Das Programm ist mit RELOAD=NO definiert, jedoch wurde es mit der Option HOLD geladen. Das Programm bleibt verfügbar, bis es durch eine andere Task freigegeben wird.

Anmerkung: In den ersten beiden aufgeführten Situationen kann die Verwendung von FREEMAIN64 möglicherweise transaktionsübergreifende Affinitäten erstellen, die sich negativ auf die Verwendung des dynamischen Transaktionsroutings auswirken. Weitere Informationen zu Transaktionsaffinitäten finden Sie unter [Affinität](#).

Sie können CICS-Schlüsselspeicher aus einem Programm nur freigeben, wenn es im CICS-Schlüssel ausgeführt wird. Wenn der Speicher aus CICS-Schlüsselspeicher abgerufen wurde und sich das Programm, das die Anforderung FREEMAIN64 absetzt, im Benutzerschlüssel befindet, tritt eine Bedingung INVREQ mit dem RESP2-Wert 2 auf.

Optionen

DATA(*datenbereich64*)

Gibt den Datenbereich des Hauptspeichers an, der freigegeben werden soll.

In Assemblersprache muss *datenbereich64* ein verschiebbarer Ausdruck sein, der eine Datenreferenz ist.

Die Länge des freigegebenen Speichers ist die Länge, die durch die ursprüngliche Anforderung abgerufen wurde und nicht unbedingt die Länge des Datenbereichs.

DATAPOINTER(*zeigerwert64*)

Gibt die Adresse des Hauptspeichers, der freigegeben werden soll, als 64-Bit-Zeigerverweis an. Bei diesem Speicher kann es sich um Speicher handeln, der durch eine vorherige Anforderung GETMAIN oder GETMAIN64 abgerufen wurde. Es kann zum Beispiel ein 64-Bit-Zeigerverweis auf einen Bereich von 31-Bit-Speicher angegeben werden.

Die Länge des freigegebenen Speichers ist die Länge, die durch die ursprüngliche Anforderung abgerufen wurde.

Bedingungen

16 INVREQ

RESP2-Werte:

1

Der Speicher, der durch den Parameter DATA oder DATAPOINTER angegeben wurde, ist kein Speicher, der durch einen Befehl GETMAIN oder GETMAIN64 abgerufen wurde.

2

Der Speicherbereich, der durch den Parameter DATA oder DATAPOINTER angegeben wurde, befindet sich im CICS-Schlüsselspeicher und das Programm, das den Befehl FREEMAIN64 abgesetzt hat, befindet sich im Benutzerschlüssel.

3

Der Speicherbereich, der durch den Parameter DATA oder DATAPOINTER angegeben wurde, wird von CICS verwaltet und jede Anforderung FREEMAIN, die versucht, CICS-verwalteten Speicherbereiche freizugeben, wird zurückgewiesen.

Beispiele: Assembler

Im folgenden Beispiel wird Speicher unter Verwendung der Option DATA freigegeben.

```
WORKAREA DS CL100
      .
      .
      EXEC CICS GETMAIN SET(9) FLENGTH(1048576)
      USING WORKAREA,9
      EXEC CICS FREEMAIN64 DATA(WORKAREA)
```

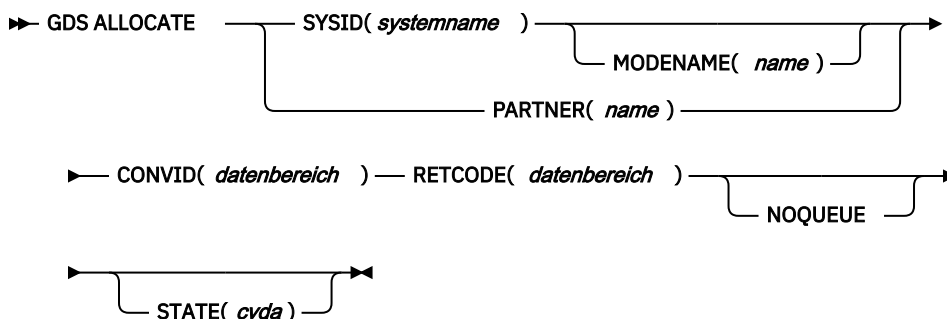
Im folgenden Beispiel wird Speicher unter Verwendung der Option DATAPOINTER freigegeben.

```
WORKAREA DS CL100
      .
      .
      EXEC CICS GETMAIN SET(9) LENGTH(100)
      USING WORKAREA,9
      .
      .
      DROP 9
      .
      EXEC CICS FREEMAIN64 DATAPOINTER(9)
```

GDS ALLOCATE

Fordert eine Sitzung mit Verbindung zu einem fernen System zur Verwendung durch einen APPC-Basisdialog an (nur Programme in Assemblersprache und C).

GDS ALLOCATE (APPC-Basisdialog)



Beschreibung

Der Befehl GDS ALLOCATE fordert eine Sitzung mit Verbindung zu einem fernen System an.

Der Rückgabecode wird in RETCODE angegeben (siehe [Tabelle 1 auf Seite 213](#)). Eine Liste der Rückgabecodewerte finden Sie unter [Rückgabecodes für APPC-Basisdialoge](#). EXEC CICS-Bedingungen werden für GDS-Befehle nie ausgelöst.

Optionen

Im Allgemeinen können die Argumente durch absolute oder verschiebbare Ausdrücke in Assemblersprache ersetzt werden.

CONVID(*datenbereich*)

Gibt den vier Zeichen langen Anwendungsdatenbereich an, der das Token aufnehmen soll, das durch einen Befehl ALLOCATE zurückgegeben wird, um den zugeordneten Dialog zu identifizieren. Dieses Token ist in nachfolgenden GDS-Befehlen erforderlich, die für den Dialog abgesetzt werden.

MODENAME(name)

Gibt den Namen der Modusgruppe an, aus der die Sitzung angefordert werden soll. Wenn Sie SYSID angeben und MODENAME nicht angeben, wählt CICS einen Modusnamen unter den für das System definierten Modusnamen aus.

NOQUEUE

Gibt an, dass die Anforderung zur Zuordnung einer Sitzung nicht in die Warteschlange gestellt werden soll, wenn eine passende APPC-Sitzung nicht sofort abgerufen werden kann. Eine Sitzung wird nur dann sofort abgerufen, wenn sie eine gebundene Konfliktgewinnersitzung ist, die noch keinem anderen Dialog zugeordnet ist.

Der Rückgabecode in RETCODE gibt an, ob eine Sitzung abgerufen wurde oder nicht.

Wenn die Option NOQUEUE nicht verwendet wird, kann eine Verzögerung auftreten, bevor die Steuerung an das Anwendungsprogramm zurückgegeben wird. Eine Verzögerung kann aus einer der folgenden Ursachen auftreten:

- Alle Sitzungen für die angegebenen Werte in SYSID und MODENAME sind im Gebrauch.
- Der CICS-Zuordnungsalgorithmus hat eine Sitzung ausgewählt, die momentan nicht gebunden ist (sodass CICS sie binden muss).
- Der CICS-Zuordnungsalgorithmus hat eine Konfliktverlierersitzung ausgewählt (sodass CICS eine Anforderung senden muss).

Falls eine Verzögerung auftritt, wartet das Programm, bis die Sitzung abgerufen wurde.

PARTNER(name)

Gibt den Namen (acht Zeichen) einer Gruppe von Definitionen an, die die Namen einer fernen logischen Einheit (NETNAME) und eines Kommunikationsprofils enthalten, die in einer zugeordneten Sitzung verwendet werden sollen. Für APPC-Basisdialoge ist MODENAME das einzige relevante Attribut, das durch Profil festgelegt wird.

Wenn Sie diese Option als Alternative zu SYSID und MODENAME verwenden, verwendet CICS den Netznamen (NETNAME) und den Modusnamen (MODENAME) aus der PARTNER-Definition.

RETCODE(datenbereich)

Gibt den sechs Byte langen Anwendungsdatenbereich an, in den Rückgabecodeinformationen (siehe [Tabelle 1 auf Seite 213](#)) verschoben werden sollen.

STATE(cvda)

Ruft den Status des aktuellen Datenaustausches ab. Die folgenden CVDA-Werte können von CICS zurückgegeben werden:

- ALLOCATED
- CONFFREE
- CONFRECEIVE
- CONFSEND
- FREE
- PENDFREE
- PENDRECEIVE
- RECEIVE
- ROLLBACK
- SEND
- SYNCFREE
- SYNCRECEIVE
- SYNCSEND

SYSID(systemname)

Gibt das ferne System an, dem eine APPC-Sitzung zugeordnet werden soll. Der Name (1-4 Zeichen) identifiziert einen Eintrag (der als APPC-Verbindung definiert ist) in der CICS-Terminalsteuertabelle (TCT).

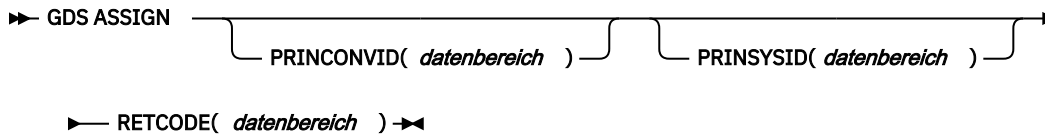
<i>Tabelle 1. Rückgabecodes für den Befehl GDS ALLOCATE</i>	
RETCODE (hexadezimal)	Beschreibung
01 0C 00	SYSID wurde nicht erkannt.
01 0C 04	SYSID ist kein LUTYPE6.2-Verbindungsname.
01 04 04	NOQUEUE wurde angegeben, jedoch sind keine gebundenen Verbindungsgewinnersitzungen verfügbar.
01 04 08	MODENAME ist nicht bekannt.
01 04 0C	Der Wert in MODENAME ist SNASVCMG, der auf die Verwendung durch CICS beschränkt ist.
01 04 0C	z/OS Communications Server hat eine Serviceklassentabelle (COS-Tabelle) für den Wert in MODENAME.
01 04 10	Die Task wurde abgebrochen, während sich der Befehl in der Warteschlange befand.
01 04 14	Alle Modusgruppen sind geschlossen.
01 04 14	Die angeforderte Modusgruppe ist geschlossen.
01 04 18	Die angeforderte Modusgruppe wird entleert (geschlossen).
01 08 00	Alle Sitzungen in der angeforderten Modusgruppe sind nicht verwendbar.
01 08 00	Die Verbindung ist im Quiescestatus.
01 08 00	Die Verbindung ist außer Betrieb.
01 08 00	Die Verbindung wurde nicht abgerufen.
01 08 00	Die lokale Maximalanzahl (zulässige Maximalanzahl von Sitzungen) ist 0.
01 08 00	Der VTAM-ACB ist geschlossen.
01 0C 14	Der Netzname (NETNAME), der in der PARTNER-Definition angegeben ist, ist nicht bekannt.
02 0C 00	PARTNER ist nicht bekannt.
06 00 00	Das Profil (PROFILE), das in der PARTNER-Definition angegeben ist, ist nicht bekannt.

Anmerkung: VTAM ist jetzt z/OS Communications Server.

GDS ASSIGN

Ruft die ID der Hauptfunktion ab, die von dem APPC-Basisdialog verwendet wird (nur Programme in Assemblersprache und C).

GDS ASSIGN (APPC-Basisdialog)



Beschreibung

Der Befehl GDS ASSIGN ruft die ID der Hauptfunktion ab.

Der Rückgabecode wird in RETCODE angegeben (siehe Tabelle 2 auf Seite 214). Eine Liste der Rückgabecodewerte finden Sie unter [Rückgabecodes für APPC-Basisdialoge](#). EXEC CICS-Bedingungen werden für GDS-Befehle nie ausgelöst.

Optionen

Im Allgemeinen können die Argumente durch absolute oder verschiebbare Ausdrücke in Assemblersprache ersetzt werden.

PRINCONVID(*datenbereich*)

Gibt einen vier Byte langen Datenbereich an, in den das Dialogtoken (CONVID) der Hauptfunktion zurückgegeben werden soll.

PRINSYSID(*datenbereich*)

Gibt einen vier Byte langen Datenbereich an, in den die System-ID (SYSID) der Hauptfunktion zurückgegeben werden soll.

RETCODE(*datenbereich*)

Gibt den sechs Byte langen Anwendungsdatenbereich an, in den Rückgabecodeinformationen (siehe Tabelle 2 auf Seite 214) verschoben werden sollen.

Tabelle 2. Rückgabecodes für den Befehl GDS ASSIGN

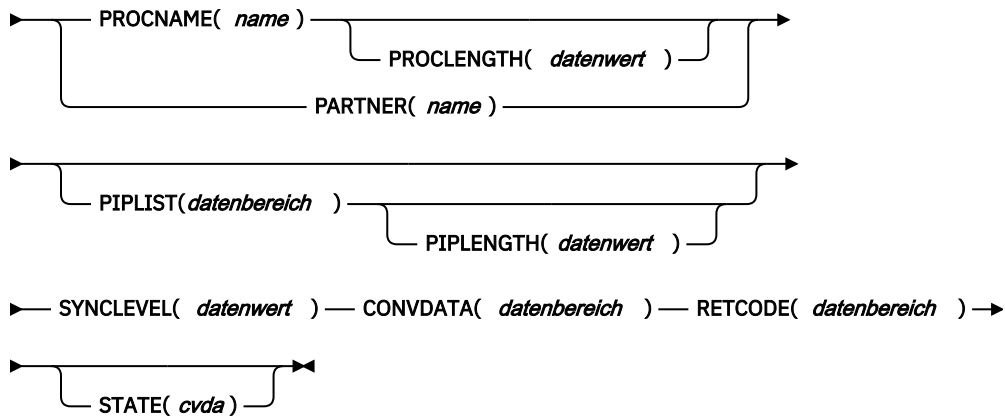
RETCODE (hexadezimal)	Beschreibung
03 00	Die Hauptfunktion ist keine APPC-Hauptfunktion.
03 04	Die Hauptfunktion ist keine APPC-Basishauptfunktion.
04	Es ist keine Terminalhauptfunktion vorhanden.

GDS CONNECT PROCESS

Initialisiert einen APPC-Basisdialog (nur Programme in Assemblersprache und C).

GDS CONNECT PROCESS (APPC-Basisdialog)

➡ GDS CONNECT PROCESS — CONVID(*name*) →



Beschreibung

EXEC CICS-Bedingungen werden für GDS-Befehle nie ausgelöst.

Der Rückgabecode wird in RETCODE angegeben (siehe [Tabelle 3 auf Seite 216](#)). Eine Liste der Rückgabecodewerte finden Sie unter [Rückgabecodes für APPC-Basisdialoge](#). EXEC CICS-Bedingungen werden für GDS-Befehle nie ausgelöst.

Mit dem Befehl GDS CONNECT PROCESS kann das Anwendungsprogramm eine Partneranwendung angeben, die auf dem fernen System ausgeführt werden soll.

Optionen

Im Allgemeinen können die Argumente durch absolute oder verschiebbare Ausdrücke in Assemblersprache ersetzt werden.

CONVDATA(datenbereich)

Gibt den 24 Byte langen Anwendungsdatenbereich an, in den dialogbezogene Informationen verschoben werden sollen. Eine Beschreibung des Formats des Datenbereichs finden Sie in den Informationen zu CONVDATA-Feldern unter [Indikatoren testen](#).

CONVID(name)

Gibt den Dialog an, auf den sich der Befehl bezieht. Der vier Zeichen lange Name bezeichnet entweder das Token, das von einem zuvor ausgeführten Befehl GDS ALLOCATE zurückgegeben wurde, oder das Token, das die Hauptsitzung (die durch einen zuvor ausgeführten Befehl GDS ASSIGN zurückgegeben wurde) darstellt.

PARTNER(name)

Gibt den Namen (8 Zeichen) einer Gruppe von Definitionen an, die den Namen (oder erweiterten Namen) einer fernen Partnertransaktion (TPNAME oder XTPNAME) einschließt. Sie können diese Option als Alternative zu PROCNAME und PROCLength verwenden.

PIPLENGTH(datenwert)

Gibt die Gesamtlänge der Liste der Prozessinitialisierungsparameter (PIP) an, die in einem Befehl CONNECT PROCESS angegeben wurde.

PIPLIST(datenbereich)

Gibt die PIP-Daten an, die an den fernen Prozess gesendet werden sollen.

PROCLength(datenwert)

Gibt die Länge des Zielprozessnamens als Halbwort-Binärwert im Bereich von 1-64 an.

PROCNAME(name)

Gibt den Namen der fernen Anwendung an. Die APPC-Architektur lässt Namen mit Längen von 1-64 Byte zu, jedoch darf jedes Produkt den eigenen Maximalwert selbst festlegen. Wenn das ferne System

ein CICS-System ist, können Sie die vier Zeichen lange Standardtransaktions-ID verwenden. Sie können auch den Wert von TPNAME in der Transaktionsdefinition verwenden.

RETCODE(datenbereich)

Gibt den sechs Byte langen Anwendungsdatenbereich an, in den Rückgabecodeinformationen (siehe Tabelle 3 auf Seite 216) verschoben werden sollen.

STATE(cvda)

Ruft den Status des aktuellen Datenaustausches ab. Die folgenden CVDA-Werte können von CICS zurückgegeben werden:

- ALLOCATED
- CONFFREE
- CONFRECEIVE
- CONFSEND
- FREE
- PENDFREE
- PENDRECEIVE
- RECEIVE
- ROLLBACK
- SEND
- SYNCFREE
- SYNCRECEIVE
- SYNCSEND

SYNCLEVEL(datenwert)

Gibt die Synchronisationsebene (Halbwort-Binärwert) an, die für den aktuellen Dialog vorgesehen ist. Die folgenden Werte sind möglich:

- 0 Keine
- 1 Bestätigen
- 2 Synchronisationspunkt

<i>Tabelle 3. Rückgabecodes für den Befehl GDS CONNECT PROCESS</i>	
RETCODE (hexadezimal)	Beschreibung
02 0C 00	PARTNER ist nicht bekannt.
03 00	CONVID ist für eine Sitzung, die keine APPC-Sitzung ist.
03 00	CONVID ist für eine Sitzung, die von CPI Communications verwendet wird.
03 04	CONVID ist für einen Dialog, der kein Basisdialog ist.
03 0C	Die Option SYNCLEVEL gibt einen anderen Wert als 0, 1 oder 2 an.
03 0C	Die Option SYNCLEVEL hat entweder 1 oder 2 angefordert, jedoch war diese Ebene nicht verfügbar.
03 08	Eine Statusprüfung hat stattgefunden.
04	CONVID ist für eine Sitzung, die der Task nicht zugeordnet ist oder die ein Relay-Link ist.
05 00 00 00 00 20	PROCLENGTH liegt außerhalb des Bereichs von 1 bis 64.
05 00 00 00 7F FF	Der Wert von PIPELENGTH liegt außerhalb des Bereichs von 4 bis 763.
05 00 00 00 7F FF	Das zwei Byte lange Längenfeld (LL) für einen der PIPs ist kleiner als 4.

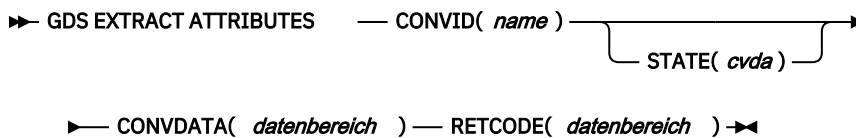
Tabelle 3. Rückgabecodes für den Befehl GDS CONNECT PROCESS (Forts.)

RETCODE (hexadezimal)	Beschreibung
05 00 00 00 7F FF	Die Gesamtlänge der LLs in den PIP-Daten ist größer als der Wert von PIPLENGTH.

GDS EXTRACT ATTRIBUTES

Greift auf die Statusinformationen eines APPC-Basisdialogs zu (nur Programme in Assemblersprache und C).

GDS EXTRACT ATTRIBUTES (APPC-Basisdialog)



Beschreibung

Der Befehl GDS EXTRACT ATTRIBUTES greift auf Statusinformationen zu einem APPC-Basisdialog zu.

Der Rückgabecode wird in RETCODE angegeben (siehe [Tabelle 4 auf Seite 218](#)). Eine Liste der Rückgabecodewerte finden Sie unter [Rückgabecodes für APPC-Basisdialoge](#). EXEC CICS-Bedingungen werden für GDS-Befehle nie ausgelöst.

Optionen

Im Allgemeinen können die Argumente durch absolute oder verschiebbare Ausdrücke in Assemblersprache ersetzt werden.

CONVID(*name*)

Gibt den Dialog an, auf den sich der Befehl bezieht. Der vier Zeichen lange Name bezeichnet entweder das Token, das von einem zuvor ausgeführten Befehl GDS ALLOCATE zurückgegeben wurde, oder das Token, das die Hauptsitzung (die durch einen zuvor ausgeführten Befehl GDS ASSIGN zurückgegeben wurde) darstellt.

CONVDATA(*datenbereich*)

Gibt den 24 Byte langen Anwendungsdatenbereich an, in den dialogbezogene Informationen verschoben werden sollen. Eine Beschreibung des Formats des Datenbereichs finden Sie in den Informationen zu CONVDATA-Feldern unter Indikatoren testen.

RETCODE(*datenbereich*)

Gibt den sechs Byte langen Anwendungsdatenbereich an, in den Rückgabecodeinformationen (siehe Tabelle 4 auf Seite 218) verschoben werden sollen.

STATE(*cvda*)

Ruft den Status des aktuellen Datenaustausches ab. Die folgenden CVDA-Werte können von CICS zurückgegeben werden:

- ALLOCATED
- CONFFREE
- CONFRECEIVE
- CONFSEND
- FREE
- PENDFREE
- PENDRECEIVE

- RECEIVE
- ROLLBACK
- SEND
- SYNCFREE
- SYNCRECEIVE
- SYNCSEND

Tabelle 4. Rückgabecodes für den Befehl GDS EXTRACT ATTRIBUTES

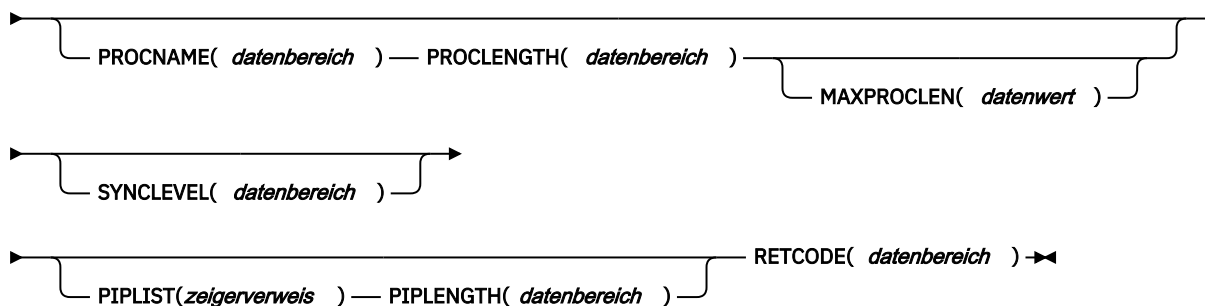
RETCODE (hexadezimal)	Beschreibung
03 00	CONVID ist für eine Sitzung, die keine APPC-Sitzung ist.
03 00	CONVID ist für eine Sitzung, die von CPI Communications verwendet wird.
03 01	INVREQ für ein DPL-Serverprogramm.
03 04	CONVID ist für einen Dialog, der kein Basisdialog ist.
04	CONVID ist für eine Sitzung, die der Task nicht zugeordnet ist oder die ein Relay-Link ist.

GDS EXTRACT PROCESS

Ruft Werte aus einem APPC-Basisdialog ab (nur Programme in Assemblersprache und C).

GDS EXTRACT PROCESS (APPC-Basisdialog)

➔ GDS EXTRACT PROCESS — CONVID(*name*) ➔



Beschreibung

Der Befehl GDS EXTRACT PROCESS ruft Werte aus einem APPC-Basisdialog ab. Die abgerufenen Daten sind nur gültig, wenn der Befehl für eine APPC-Basishauptfunktion abgesetzt wird.

Der Rückgabecode wird in RETCODE angegeben (siehe [Tabelle 5 auf Seite 219](#)). Eine Liste der Rückgabecodewerte finden Sie unter [Rückgabecodes für APPC-Basisdialoge](#). EXEC CICS-Bedingungen werden für GDS-Befehle nie ausgelöst.

Optionen

Im Allgemeinen können die Argumente durch absolute oder verschiebbare Ausdrücke in Assemblersprache ersetzt werden.

CONVID(name)

Gibt den Dialog an, auf den sich der Befehl bezieht. Der vier Zeichen lange Name identifiziert das Token, das die Hauptsitzung (die durch einen zuvor ausgeführten Befehl GDS ASSIGN zurückgegeben wurde) darstellt.

MAXPROCLEN(datenwert)

Gibt die Länge (1-64 Zeichen) des PROCNAME-Datenbereichs an. Wenn die Option MAXPROCLEN nicht angegeben wird, wird eine Pufferlänge von 32 Byte angenommen.

PIPLENGTH(datenbereich)

Gibt einen Halbwort-Binärwert-Datenbereich an, der die Länge der PIP-Liste (PIPLIST) empfangen soll, die durch einen Befehl GDS EXTRACT PROCESS empfangen wurde.

PIPLIST(zeigerverweis)

Gibt einen Zeigerverweis an, der auf die Adresse der PIP-Liste gesetzt werden soll, die durch einen Befehl GDS EXTRACT PROCESS empfangen wurde. Der Wert null gibt an, dass keine PIP-Liste empfangen wurde.

PROCLENGTH(datenbereich)

Gibt einen Halbwort-Binärwert-Datenbereich an, der auf die tatsächliche Länge des Prozessnamens gesetzt wird.

PROCNAME(datenbereich)

Gibt den Zielanwendungsdatenbereich (1-64 Byte) an, in den der Prozessname verschoben werden soll, der in dem APPC-Funktionsverwaltungsheader für den Anschluss (ATTACH) angegeben wurde. Der Bereich wird mit Leerzeichen aufgefüllt, wenn dies erforderlich ist.

RETCODE(datenbereich)

Gibt den sechs Byte langen Anwendungsdatenbereich an, in den Rückgabecodeinformationen (siehe [Tabelle 5 auf Seite 219](#)) verschoben werden sollen.

SYNLEVEL(datenbereich)

Gibt einen Halbwort-Binärwert-Datenbereich an, der festgelegt wird, um die Synchronisationsebene anzugeben, die für den aktuellen Dialog wirksam ist. Die folgenden Werte sind möglich:

- 0 Keine
- 1 Bestätigen
- 2 Synchronisationspunkt

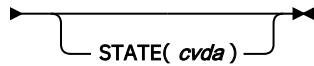
<i>Tabelle 5. Rückgabecodes für den Befehl GDS EXTRACT PROCESS</i>	
RETCODE (hexadezimal)	Beschreibung
03 00	CONVID ist für eine Sitzung, die keine APPC-Sitzung ist.
03 00	CONVID ist für eine Sitzung, die von CPI Communications verwendet wird.
03 00	CONVID ist für eine Sitzung, die nicht die Hauptfunktion ist.
03 00	Die Hauptfunktion wurde nicht durch Terminaldaten gestartet.
03 04	CONVID ist für einen Dialog, der kein Basisdialog ist.
04	CONVID ist für eine Sitzung, die der Task nicht zugeordnet ist oder die ein Relay-Link ist.
05 00 00 00 00 20	Der zurückgegebene Wert für PROCLENGTH ist größer als der Wert für MAXPROCLEN.

GDS FREE

Gibt eine APPC-Sitzung an CICS zurück (nur Programme in Assemblersprache und C).

GDS FREE (APPC-Basisdialog)

➡ GDS FREE — CONVID(*name*) — CONVDATA(*datenbereich*) — RETCODE(*datenbereich*) →



Beschreibung

Der Befehl GDS FREE gibt eine Sitzung an CICS zurück. Das Absetzen dieses Befehls ist nur gültig, wenn der Dialog beendet ist, das heißt, wenn der Dialog den Status frei (FREE) hat.

Der Rückgabecode wird in RETCODE angegeben (siehe [Tabelle 6 auf Seite 221](#)). Eine Liste der Rückgabecodewerte finden Sie unter [Rückgabecodes für APPC-Basisdialoge](#). EXEC CICS-Bedingungen werden für GDS-Befehle nie ausgelöst.

Optionen

Im Allgemeinen können die Argumente durch absolute oder verschiebbare Ausdrücke in Assemblersprache ersetzt werden.

CONVDATA(*datenbereich*)

Gibt den 24 Byte langen Anwendungsdatenbereich an, in den dialogbezogene Informationen verschoben werden sollen. Eine Beschreibung des Formats des Datenbereichs finden Sie in den Informationen zu CONVDATA-Feldern unter [Indikatoren testen](#).

CONVID(*name*)

Gibt den Dialog an, der freigegeben werden soll. Der vier Zeichen lange Name bezeichnet entweder das Token, das von einem zuvor ausgeführten Befehl GDS ALLOCATE zurückgegeben wurde, oder das Token, das die Hauptsitzung (die durch einen zuvor ausgeführten Befehl GDS ASSIGN zurückgegeben wurde) darstellt.

RETCODE(*datenbereich*)

Gibt den sechs Byte langen Anwendungsdatenbereich an, in den Rückgabecodeinformationen (siehe [Tabelle 6 auf Seite 221](#)) verschoben werden sollen.

STATE(*cvda*)

Ruft den Status des aktuellen Datenaustausches ab. Die Option STATE für einen Befehl FREE gibt den CVDA-Code 00 zurück, wenn kein aktiver Dialog mehr vorhanden ist. Die weiteren CVDA-Ausgabewerte sind folgende:

- ALLOCATED
- CONFFREE
- CONFRECEIVE
- CONFSEND
- FREE
- PENDFREE
- PENDRECEIVE
- RECEIVE
- ROLLBACK
- SEND
- SYNCFREE
- SYNCRECEIVE
- SYNCSEND

Tabelle 6. Rückgabecodes für den Befehl GDS FREE

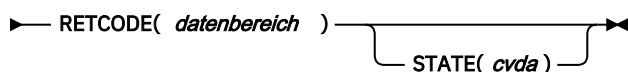
RETCODE (hexadezimal)	Beschreibung
03 00	CONVID ist für eine Sitzung, die keine APPC-Sitzung ist.
03 00	CONVID ist für eine Sitzung, die von CPI Communications verwendet wird.
03 04	CONVID ist für einen Dialog, der kein Basisdialog ist.
03 08	Eine Statusprüfung hat stattgefunden.
04	CONVID ist für eine Sitzung, die der Task nicht zugeordnet ist oder die ein Relay-Link ist.

GDS ISSUE ABEND

Führt eine abnormale Beendigung eines APPC-Basisdialogs aus (nur Programme in Assemblersprache und C).

GDS ISSUE ABEND (APPC-Basisdialog)

➔ GDS ISSUE ABEND — CONVID(*name*) — CONVDATA(*datenbereich*) ➔



Beschreibung

Der Befehl GDS ISSUE ABEND bewirkt völlig unabhängig vom Dialogstatus eine unverzügliche Beendigung eines APPC-Basisdialogs. Die Partnertransaktion wird informiert.

Der Rückgabecode wird in RETCODE angegeben (siehe Tabelle 7 auf Seite 222). Eine Liste der Rückgabecodewerte finden Sie unter [Rückgabecodes für APPC-Basisdialoge](#). EXEC CICS-Bedingungen werden für GDS-Befehle nie ausgelöst.

Optionen

Im Allgemeinen können die Argumente durch absolute oder verschiebbare Ausdrücke in Assemblersprache ersetzt werden.

CONVDATA(*datenbereich*)

Gibt den 24 Byte langen Anwendungsdatenbereich an, in den dialogbezogene Informationen verschoben werden sollen. Eine Beschreibung des Formats des Datenbereichs finden Sie in den Informationen zu CONVDATA-Feldern unter [Indikatoren testen](#).

CONVID(*name*)

Gibt den Dialog an, auf den sich der Befehl bezieht. Der vier Zeichen lange Name bezeichnet entweder das Token, das von einem zuvor ausgeführten Befehl GDS ALLOCATE zurückgegeben wurde, oder das Token, das die Hauptsitzung (die durch einen zuvor ausgeführten Befehl GDS ASSIGN zurückgegeben wurde) darstellt.

RETCODE(*datenbereich*)

Gibt den sechs Byte langen Anwendungsdatenbereich an, in den Rückgabecodeinformationen (siehe Tabelle 7 auf Seite 222) verschoben werden sollen.

STATE(*cvda*)

Ruft den Status des aktuellen Datenaustausches ab. Die folgenden CVDA-Werte können von CICS zurückgegeben werden:

- ALLOCATED
- CONFFREE

- CONFRECEIVE
- CONFSEND
- FREE
- PENDFREE
- PENDRECEIVE
- RECEIVE
- ROLLBACK
- SEND
- SYNCFREE
- SYNCRECEIVE
- SYNCSEND

Tabelle 7. Rückgabecodes für den Befehl GDS ISSUE ABEND

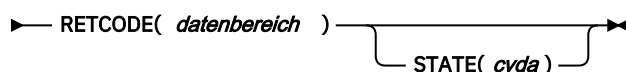
RETCODE (hexadezimal)	Beschreibung
03 00	CONVID ist für eine Sitzung, die keine APPC-Sitzung ist.
03 00	CONVID ist für eine Sitzung, die von CPI Communications verwendet wird.
03 04	CONVID ist für einen Dialog, der kein Basisdialog ist.
03 08	Eine Statusprüfung hat stattgefunden.
04	CONVID ist für eine Sitzung, die der Task nicht zugeordnet ist oder die ein Relay-Link ist.

GDS ISSUE CONFIRMATION

Setzt eine Synchronisationsanforderung für einen APPC-Basisdialog ab (nur Programme in Assemblersprache und C).

GDS ISSUE CONFIRMATION (APPC-Basisdialog)

► GDS ISSUE CONFIRMATION — CONVID(*name*) — CONVDATA(*datenbereich*) →



Beschreibung

Der Befehl GDS ISSUE CONFIRMATION setzt eine Synchronisationsanforderung als Antwort auf einen Befehl GDS SEND CONFIRM ab, der von einer Partnertransaktion abgesetzt wurde.

Der Rückgabecode wird in RETCODE angegeben (siehe [Tabelle 8 auf Seite 223](#)). Eine Liste der Rückgabecodewerte finden Sie unter [Rückgabecodes für APPC-Basisdialoge](#). EXEC CICS-Bedingungen werden für GDS-Befehle nie ausgelöst.

Optionen

Im Allgemeinen können die Argumente durch absolute oder verschiebbare Ausdrücke in Assemblersprache ersetzt werden.

CONVDATA(*datenbereich*)

Gibt den 24 Byte langen Anwendungsdatenbereich an, in den dialogbezogene Informationen verschoben werden sollen. Eine Beschreibung des Formats des Datenbereichs finden Sie in den Informationen zu CONVDATA-Feldern unter [Indikatoren testen](#).

CONVID(name)

Gibt den Dialog an, auf den sich der Befehl bezieht. Der vier Zeichen lange Name bezeichnet entweder das Token, das von einem zuvor ausgeführten Befehl GDS ALLOCATE zurückgegeben wurde, oder das Token, das die Hauptsitzung (die durch einen zuvor ausgeführten Befehl GDS ASSIGN zurückgegeben wurde) darstellt.

RETCODE(datenbereich)

Gibt den sechs Byte langen Anwendungsdatenbereich an, in den Rückgabecodeinformationen (siehe Tabelle 8 auf Seite 223) verschoben werden sollen.

STATE(cvda)

Ruft den Status des aktuellen Datenaustausches ab. Die folgenden CVDA-Werte können von CICS zurückgegeben werden:

- ALLOCATED
- CONFFREE
- CONFRECEIVE
- CONFSEND
- FREE
- PENDFREE
- PENDRECEIVE
- RECEIVE
- ROLLBACK
- SEND
- SYNCFREE
- SYNCRECEIVE
- SYNCSEND

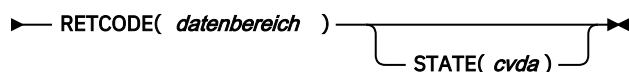
Tabelle 8. Rückgabecodes für den Befehl GDS ISSUE CONFIRMATION	
RETCODE (hexadezimal)	Beschreibung
03 00	CONVID ist für eine Sitzung, die keine APPC-Sitzung ist.
03 00	CONVID ist für eine Sitzung, die von CPI Communications verwendet wird.
03 04	CONVID ist für einen Dialog, der kein Basisdialog ist.
03 08	Eine Statusprüfung hat stattgefunden.
03 14	Der Befehl wurde für einen Dialog der Synchronisationsebene 0 abgesetzt.
04	CONVID ist für eine Sitzung, die der Task nicht zugeordnet ist oder die ein Relay-Link ist.

GDS ISSUE ERROR

Informiert einen APPC-Basisdialogpartner über einen Fehler (nur Programme in Assemblersprache und C).

GDS ISSUE ERROR (APPC-Basisdialog)

➡ GDS ISSUE ERROR — CONVID(*name*) — CONVDATA(*datenbereich*) →



Beschreibung

Der Befehl GDS ISSUE ERROR informiert den Dialogpartner, dass ein Fehler aufgetreten ist.

Der Rückgabecode wird weiter unten in RETCODE angegeben. Eine Liste der Rückgabecodewerte finden Sie unter [Rückgabecodes für APPC-Basisdialoge](#). EXEC CICS-Bedingungen werden für GDS-Befehle nie ausgelöst.

Optionen

Im Allgemeinen können die Argumente durch absolute oder verschiebbare Ausdrücke in Assemblersprache ersetzt werden.

CONVDATA(*datenbereich*)

Gibt den 24 Byte langen Anwendungsdatenbereich an, in den dialogbezogene Informationen verschoben werden sollen. Eine Beschreibung des Formats des Datenbereichs finden Sie in den Informationen zu CONVDATA-Feldern unter [Indikatoren testen](#).

CONVID(*name*)

Gibt den Dialog an, auf den sich der Befehl bezieht. Der vier Zeichen lange Name bezeichnet entweder das Token, das von einem zuvor ausgeführten Befehl GDS ALLOCATE zurückgegeben wurde, oder das Token, das die Hauptsitzung (die durch einen zuvor ausgeführten Befehl GDS ASSIGN zurückgegeben wurde) darstellt.

RETCODE(*datenbereich*)

Gibt den sechs Byte langen Anwendungsdatenbereich an, in den Rückgabecodeinformationen (siehe [Tabelle 9 auf Seite 224](#)) verschoben werden sollen.

STATE(*cvda*)

Ruft den Status des aktuellen Datenaustausches ab. Die folgenden CVDA-Werte können von CICS zurückgegeben werden:

- ALLOCATED
- CONFFREE
- CONFRECEIVE
- CONFSEND
- FREE
- PENDFREE
- PENDRECEIVE
- RECEIVE
- ROLLBACK
- SEND
- SYNCFREE
- SYNCRECEIVE
- SYNCSEND

Tabelle 9. Rückgabecodes für den Befehl GDS ISSUE ERROR	
RETCODE (hexadezimal)	Beschreibung
03 00	CONVID ist für eine Sitzung, die keine APPC-Sitzung ist.
03 00	CONVID ist für eine Sitzung, die von CPI Communications verwendet wird.
03 04	CONVID ist für einen Dialog, der kein Basisdialog ist.
03 08	Eine Statusprüfung hat stattgefunden.

Tabelle 9. Rückgabecodes für den Befehl GDS ISSUE ERROR (Forts.)

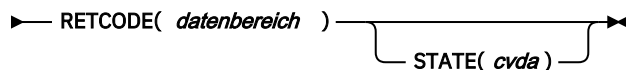
RETCODE (hexadezimal)	Beschreibung
04	CONVID ist für eine Sitzung, die der Task nicht zugeordnet ist oder die ein Relay-Link ist.

GDS ISSUE PREPARE

Setzt den ersten Ablauf einer Synchronisationspunktanforderung für einen APPC-Basisdialog ab (nur Programme in Assemblersprache und C).

GDS ISSUE PREPARE (APPC-Basisdialog)

►► GDS ISSUE PREPARE — CONVID(*name*) — CONVDATA(*datenbereich*) →



Beschreibung

Der Befehl GDS ISSUE PREPARE setzt den ersten Ablauf einer Synchronisationspunktanforderung für einen APPC-Basisdialog ab.

Der Rückgabecode wird in RETCODE angegeben (siehe Tabelle 10 auf Seite 226). Eine Liste der Rückgabecodewerte finden Sie unter [Rückgabecodes für APPC-Basisdialoge](#). EXEC CICS-Bedingungen werden für GDS-Befehle nie ausgelöst.

Optionen

Im Allgemeinen können die Argumente durch absolute oder verschiebbare Ausdrücke in Assemblersprache ersetzt werden.

CONVDATA(*datenbereich*)

Gibt den 24 Byte langen Anwendungsdatenbereich an, in den dialogbezogene Informationen verschoben werden sollen. Eine Beschreibung des Formats des Datenbereichs finden Sie in den Informationen zu CONVDATA-Feldern unter [Indikatoren testen](#).

CONVID(*name*)

Gibt den Dialog an, auf den sich der Befehl bezieht. Der vier Zeichen lange Name bezeichnet entweder das Token, das von einem zuvor ausgeführten Befehl GDS ALLOCATE zurückgegeben wurde, oder das Token, das die Hauptsitzung (die durch einen zuvor ausgeführten Befehl GDS ASSIGN zurückgegeben wurde) darstellt.

RETCODE(*datenbereich*)

Gibt den sechs Byte langen Anwendungsdatenbereich an, in den Rückgabecodeinformationen (siehe Tabelle 10 auf Seite 226) verschoben werden sollen.

STATE(*cvda*)

Ruft den Status des aktuellen Datenaustausches ab. Die folgenden CVDA-Werte können von CICS zurückgegeben werden:

- ALLOCATED
- CONFFREE
- CONFRECEIVE
- CONFSEND
- FREE
- PENDFREE
- PENDRECEIVE

- RECEIVE
- ROLLBACK
- SEND
- SYNCFREE
- SYNCRECEIVE
- SYNCSEND

Tabelle 10. Rückgabecodes für den Befehl GDS ISSUE PREPARE

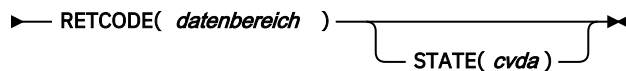
RETCODE (hexadezimal)	Beschreibung
03 00	CONVID ist für eine Sitzung, die keine APPC-Sitzung ist.
03 00	CONVID ist für eine Sitzung, die von CPI Communications verwendet wird.
03 04	CONVID ist für einen Dialog, der kein Basisdialog ist.
03 0C	Der Befehl wurde für einen Dialog abgesetzt, der nicht Synchronisationsebene 2 (Sync-Level 2) hat.
03 24	Ein Statusfehler ist aufgetreten.
04	CONVID ist für eine Sitzung, die der Task nicht zugeordnet ist oder die ein Relay-Link ist.

GDS ISSUE SIGNAL

Fordert einen Richtungswechsel vom sendenden APPC-Basisdialog für die Transaktion an (nur Programme in Assemblersprache und C).

GDS ISSUE SIGNAL (APPC-Basisdialog)

➔ GDS ISSUE SIGNAL — CONVID(*name*) — CONVDATA(*datenbereich*) ➔



Beschreibung

Der Befehl GDS ISSUE SIGNAL fordert einen Richtungswechsel an.

Der Rückgabecode wird in RETCODE angegeben (siehe [Tabelle 11 auf Seite 227](#)). Eine Liste der Rückgabecodewerte finden Sie unter [Rückgabecodes für APPC-Basisdialoge](#). EXEC CICS-Bedingungen werden für GDS-Befehle nie ausgelöst.

Optionen

Im Allgemeinen können die Argumente durch absolute oder verschiebbare Ausdrücke in Assemblersprache ersetzt werden.

CONVDATA(*datenbereich*)

Gibt den 24 Byte langen Anwendungsdatenbereich an, in den dialogbezogene Informationen verschoben werden sollen. Eine Beschreibung des Formats des Datenbereichs finden Sie in den Informationen zu CONVDATA-Feldern unter [Indikatoren testen](#).

CONVID(*name*)

Gibt den Dialog an, auf den sich der Befehl bezieht. Der vier Zeichen lange Name bezeichnet entweder das Token, das von einem zuvor ausgeführten Befehl GDS ALLOCATE zurückgegeben wurde, oder das Token, das die Hauptsitzung (die durch einen zuvor ausgeführten Befehl GDS ASSIGN zurückgegeben wurde) darstellt.

RETCODE(*datenbereich*)

Gibt den sechs Byte langen Anwendungsdatenbereich an, in den Rückgabecodeinformationen (siehe Tabelle 11 auf Seite 227) verschoben werden sollen.

STATE(*cvda*)

Ruft den Status des aktuellen Datenaustausches ab. Die folgenden CVDA-Werte können von CICS zurückgegeben werden:

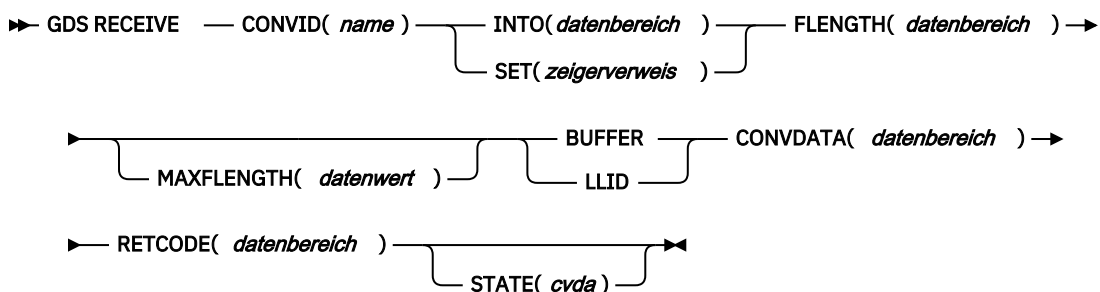
- ALLOCATED
- CONFFREE
- CONFRECEIVE
- CONFSEND
- FREE
- PENDFREE
- PENDRECEIVE
- RECEIVE
- ROLLBACK
- SEND
- SYNCFREE
- SYNCRECEIVE
- SYNCSEND

Tabelle 11. Rückgabecodes für den Befehl GDS ISSUE SIGNAL	
RETCODE (hexadezimal)	Beschreibung
03 00	CONVID ist für eine Sitzung, die keine APPC-Sitzung ist.
03 00	CONVID ist für eine Sitzung, die von CPI Communications verwendet wird.
03 04	CONVID ist für einen Dialog, der kein Basisdialog ist.
03 08	Eine Statusprüfung hat stattgefunden.
04	CONVID ist für eine Sitzung, die der Task nicht zugeordnet ist oder die ein Relay-Link ist.

GDS RECEIVE

Empfängt Daten aus einem APPC-Basisdialog (nur Assemblersprachen- und C-Programme).

GDS RECEIVE (APPC-Basisdialog)



Beschreibung

Der Befehl GDS RECEIVE empfängt Daten und Anzeiger von einer Partnertransaktion.

Der Rückgabecode wird in RETCODE angegeben (siehe Tabelle 12 auf Seite 229). Eine Liste der Rückgabecodewerte finden Sie unter [Rückgabecodes für APPC-Basisdialoge](#). EXEC CICS-Bedingungen werden für GDS-Befehle nie ausgelöst.

Optionen

Im Allgemeinen können die Argumente durch absolute oder verschiebbare Ausdrücke in Assemblersprache ersetzt werden.

BUFFER

Gibt an, dass die Länge der Daten, die an das Anwendungsprogramm als Antwort auf den Befehl RECEIVE übergeben werden, nur durch die Länge, die in der Option MAXLENGTH angegeben wird, beschränkt werden sollen, und nicht von Grenzen strukturierter GDS-Felder betroffen sein sollen. Die Steuerung wird an das Anwendungsprogramm zurückgegeben, wenn diese Länge empfangen wurde oder wenn eine Synchronisationsanforderung, eine Richtungsänderungsanforderung oder eine Klammerendeanforderung (Bracket-End) empfangen wird.

CONVDATA(datenbereich)

Gibt den 24 Byte langen Anwendungsdatenbereich an, in den dialogbezogene Informationen verschoben werden sollen. Eine Beschreibung des Formats des Datenbereichs finden Sie in den Informationen zu CONVDATA-Feldern unter [Indikatoren testen](#).

CONVID(name)

Gibt den Dialog an, auf den sich der Befehl bezieht. Der vier Zeichen lange Name bezeichnet entweder das Token, das von einem zuvor ausgeführten Befehl GDS ALLOCATE zurückgegeben wurde, oder das Token, das die Hauptsitzung (die durch einen zuvor ausgeführten Befehl GDS ASSIGN zurückgegeben wurde) darstellt.

FLENGTH(datenbereich)

Gibt einen Halbwort-Binärwert-Datenbereich an, der auf die Länge der Daten gesetzt wird, die für das Anwendungsprogramm verfügbar gemacht wurden.

INTO(datenbereich)

Gibt den Zieldatenbereich der Anwendung an, in den Daten aus dem Anwendungsprogramm empfangen werden sollen, das mit dem anderen Ende des aktuellen Dialogs verbunden ist. Die Länge dieses Bereichs darf nicht kleiner als der Wert sein, der in der Option MAXLENGTH wird.

LLID

Gibt an, dass die Begrenzung, die von CICS zum Beenden der Übergabe von Daten an das Anwendungsprogramm verwendet werden soll, das Ende des strukturierten GDS-Felds ist, wenn dieses angetroffen wird, bevor die Begrenzung von MAXLENGTH erreicht wird.

MAXLENGTH(datenwert)

Gibt als Vollwort-Binärwert entweder die Länge des Zieldatenbereichs, der in der Option INTO angegeben wird, oder die maximale Länge der Daten an, die durch den Zeigerverweis in der Option SET adressiert werden sollen. Die Länge darf 32 767 Byte nicht überschreiten. CICS empfängt nicht mehr Daten, als durch den Wert von MAXLENGTH zugelassen werden.

RETCODE(datenbereich)

Gibt den sechs Byte langen Anwendungsdatenbereich an, in den Rückgabecodeinformationen (siehe Tabelle 12 auf Seite 229) verschoben werden sollen.

SET(zeigerverweis)

Gibt den Zeigerverweis an, der auf die Adresse der Daten gesetzt werden soll, die aus dem Anwendungsprogramm empfangen wurden, das mit dem anderen Ende des aktuellen Dialogs verbunden ist. Der Zeigerverweis bleibt bis zum nächsten Befehl RECEIVE (GDS oder APPC) oder bis zum Ende der Task gültig, sofern er nicht durch andere Befehle oder Anweisungen geändert wird.

Wenn dem Anwendungsprogramm DATALLOCATION(ANY) zugeordnet ist, kann die Adresse der Daten ober- oder unterhalb der 16-MB-Grenze liegen.

Wenn dem Anwendungsprogramm DATALLOCATION(BELOW) zugeordnet ist und sich die Daten oberhalb der 16-MB-Grenze befinden, werden die Daten unter die 16-MB-Grenze kopiert und die Adresse dieser Kopie wird zurückgegeben.

Wenn TASKDATAKEY(USER) für die aktive Task angegeben wird und der Speicherschutz aktiv ist, werden die Daten in einem Benutzerschlüssel zurückgegeben. Wenn für die aktive Task TASKDATAKEY(CICS) angegeben wird und der Speicherschutz aktiv ist, befinden sich die zurückgegebenen Daten in einem CICS-Schlüssel.

STATE(*cvda*)

Ruft den Status des aktuellen Datenaustausches ab. Die folgenden CVDA-Werte können von CICS zurückgegeben werden:

- ALLOCATED
- CONFFREE
- CONFRECEIVE
- CONFSEND
- FREE
- PENDFREE
- PENDRECEIVE
- RECEIVE
- ROLLBACK
- SEND
- SYNCFREE
- SYNCRECEIVE
- SYNCSEND

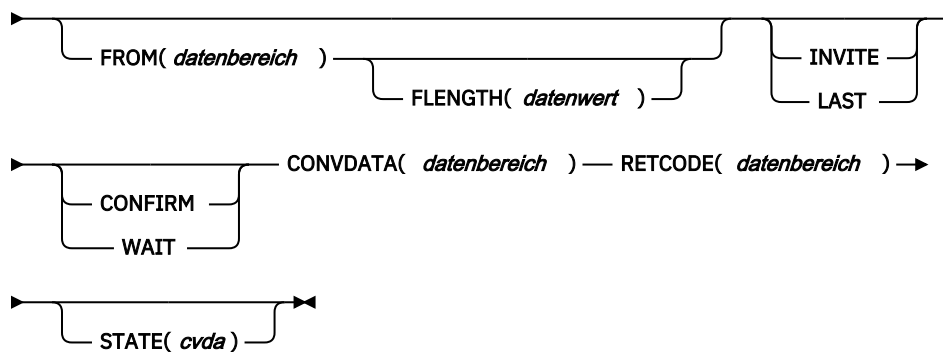
<i>Tabelle 12. Rückgabecodes für den Befehl GDS RECEIVE</i>	
RETCODE (hexadezimal)	Beschreibung
03 00	CONVID ist für eine Sitzung, die keine APPC-Sitzung ist.
03 00	CONVID ist für eine Sitzung, die von CPI Communications verwendet wird.
03 04	CONVID ist für einen Dialog, der kein Basisdialog ist.
03 08	Eine Statusprüfung hat stattgefunden.
04	CONVID ist für eine Sitzung, die der Task nicht zugeordnet ist oder die ein Relay-Link ist.
05 00 00 00 7F FF	Der Wert für MAXLENGTH liegt außerhalb des Bereichs von 0 bis 32 767.

GDS SEND

Sendet Daten an einen APPC-Basisdialog (nur Programme in Assemblersprache und C).

GDS SEND (APPC-Basisdialog)

➔ GDS SEND — CONVID(*name*) ➔



Beschreibung

Der Befehl GDS SEND sendet Daten.

Der Rückgabecode wird in RETCODE angegeben (siehe Tabelle 13 auf Seite 231). Eine Liste der Rückgabecodewerte finden Sie unter [Rückgabecodes für APPC-Basisdialoge](#). EXEC CICS-Bedingungen werden für GDS-Befehle nie ausgelöst.

Optionen

Im Allgemeinen können die Argumente durch absolute oder verschiebbare Ausdrücke in Assemblersprache ersetzt werden.

CONFIRM

Ermöglicht einer Anwendung, die auf Synchronisationsebene 1 oder 2 aktiv ist, eine Synchronisation ihrer Verarbeitung mit der eines Prozesses in einem fernen System. Die Aktionen, die zur Synchronisation der Verarbeitung ausgeführt werden, werden durch die beteiligten Anwendungsprogramme definiert. Die Option CONFIRM sorgt dafür, dass RQD2 den Daten hinzugefügt wird, die bereits gesendet wurde, und erzwingt ein Warten (WAIT). Bei Empfang des Anzeigers führt der ferne Prozess die abgestimmten Aktionen aus und sendet eine Antwort. Wenn das Warten abgeschlossen ist, wird CDBERR auf den Wert X'00' gesetzt, wenn die entsprechende Antwort empfangen wurde.

CONVDATA(datenbereich)

Gibt den 24 Byte langen Anwendungsdatenbereich an, in den dialogbezogene Informationen verschoben werden sollen. Eine Beschreibung des Formats des Datenbereichs finden Sie in den Informationen zu CONVDATA-Feldern unter [Indikatoren testen](#).

CONVID(name)

Gibt den Dialog an, auf den sich der Befehl bezieht. Der vier Zeichen lange Name bezeichnet entweder das Token, das von einem zuvor ausgeführten Befehl GDS ALLOCATE zurückgegeben wurde, oder das Token, das die Hauptsitzung (die durch einen zuvor ausgeführten Befehl GDS ASSIGN zurückgegeben wurde) darstellt.

FLENGTH(datenwert)

Gibt die Länge der Daten, die in der Option FROM angegeben wurden, als Vollwort-Binärwert aus dem Bereich von 1 bis 32 767 an.

FROM(datenbereich)

Gibt die Daten an, die gesendet werden sollen.

INVITE

Mit dieser Option kann ein Anwendungsprogramm einen Anzeiger für Änderung der Übertragungsrichtung zu Daten hinzufügen, die bereits an einen Prozess in einem verbundenen APPC-System gesendet wurden. Steuerdaten werden von CICS erst bei der nachfolgenden Ausführung eines Befehls WAIT oder SYNCPOINT übertragen, wenn nicht die Option CONFIRM oder WAIT auch im Befehl GDS SEND INVITE codiert wird.

LAST

Mit dieser Option kann ein Anwendungsprogramm CEB zu Daten hinzufügen, die bereits an einen Prozess in einem verbundenen APPC-System gesendet wurden. CEB wird von CICS erst bei der nachfolgenden Ausführung eines Befehls WAIT oder SYNCPOINT übertragen, wenn nicht die Option CONFIRM oder WAIT auch im Befehl GDS SEND LAST codiert wird. Beachten Sie, dass der Dialog in der Klammer (Bracket) bleibt, wenn einer dieser Befehle aufgrund eines dialogbezogenen Fehlers fehlschlägt. In einem solchen Fall sollte das Anwendungsprogramm einen Befehl GDS RECEIVE ausführen. Allerdings hat ein Befehl GDS SEND LAST WAIT (ohne Daten) immer zur Folge, dass der Dialog freigegeben wird.

RETCODE(*datenbereich*)

Gibt den sechs Byte langen Anwendungsdatenbereich an, in den Rückgabecodeinformationen (siehe [Tabelle 13 auf Seite 231](#)) verschoben werden sollen.

STATE(*cvda*)

Ruft den Status des aktuellen Datenaustausches ab. Die folgenden CVDA-Werte können von CICS zurückgegeben werden:

- ALLOCATED
- CONFFREE
- CONFRECEIVE
- CONFSEND
- FREE
- PENDFREE
- PENDRECEIVE
- RECEIVE
- ROLLBACK
- SEND
- SYNCFREE
- SYNCRECEIVE
- SYNCSEND

WAIT

Stellt sicher, dass alle Daten und Anzeiger, die bislang in einem Dialog gesendet wurden, aus der Partnertransaktion gelöscht werden.

Wenn die Option WAIT nicht verwendet wird, werden Daten aus aufeinanderfolgenden Befehlen SEND von CICS zusammen mit allen Anzeigern in einem internen Puffer kumuliert. Wenn der Puffer voll wird, werden die meisten der kumulierten Daten an das ferne System übertragen, nicht jedoch die kumulierten Anzeiger. Die Übertragung der kumulierten Daten und der Anzeiger wird durch die Optionen WAIT oder CONFIRM des Befehls GDS SEND oder durch den Befehl GDS WAIT erzwungen.

Tabelle 13. Rückgabecodes für den Befehl GDS SEND	
RETCODE (hexadezimal)	Beschreibung
03 00	CONVID ist für eine Sitzung, die keine APPC-Sitzung ist.
03 00	CONVID ist für eine Sitzung, die von CPI Communications verwendet wird.
03 04	CONVID ist für einen Dialog, der kein Basisdialog ist.
03 08	Eine Statusprüfung hat stattgefunden.
03 14	Die Option CONFIRM wurde für einen Dialog der Synchronisationsebene 0 verwendet.
03 10	LL-Fehler (falsch oder unvollständig).

Tabelle 13. Rückgabecodes für den Befehl GDS SEND (Forts.)

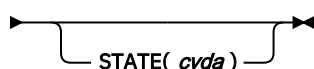
RETCODE (hexadezimal)	Beschreibung
04	CONVID ist für eine Sitzung, die der Task nicht zugeordnet ist oder die ein Relay-Link ist.
05 00 00 00 7F FF	Der Wert für FLENGTH liegt außerhalb des Bereichs von 0 bis 32 767.

GDS WAIT

Stellt sicher, dass kumulierte Daten in einem APPC-Dialog übertragen wurden (nur Programme in Assemblersprache und C).

GDS WAIT (APPC-Basisdialog)

➔ GDS WAIT — CONVID(*name*) — CONVDATA(*datenbereich*) — RETCODE(*datenbereich*) ➔



Beschreibung

Der Befehl GDS WAIT stellt sicher, dass die kumulierten Daten gesendet wurden.

Der Rückgabecode wird in RETCODE angegeben (siehe Tabelle 14 auf Seite 233). Eine Liste der Rückgabecodewerte finden Sie unter [Rückgabecodes für APPC-Basisdialoge](#). EXEC CICS-Bedingungen werden für GDS-Befehle nie ausgelöst.

Optionen

Im Allgemeinen können die Argumente durch absolute oder verschiebbare Ausdrücke in Assemblersprache ersetzt werden.

CONVDATA(*datenbereich*)

Gibt den 24 Byte langen Anwendungsdatenbereich an, in den dialogbezogene Informationen verschoben werden sollen. Eine Beschreibung des Formats des Datenbereichs finden Sie in den Informationen zu CONVDATA-Feldern unter [Indikatoren testen](#).

CONVID(*name*)

Gibt den Dialog an, auf den sich der Befehl bezieht. Der vier Zeichen lange Name bezeichnet entweder das Token, das von einem zuvor ausgeführten Befehl GDS ALLOCATE zurückgegeben wurde, oder das Token, das die Hauptsitzung (die durch einen zuvor ausgeführten Befehl GDS ASSIGN zurückgegeben wurde) darstellt.

RETCODE(*datenbereich*)

Gibt den sechs Byte langen Anwendungsdatenbereich an, in den Rückgabecodeinformationen (siehe Tabelle 14 auf Seite 233) verschoben werden sollen.

STATE(*cvda*)

Ruft den Status des aktuellen Datenaustausches ab. Die folgenden CVDA-Werte können von CICS zurückgegeben werden:

- ALLOCATED
- CONFFREE
- CONFRECEIVE
- CONFSEND
- FREE
- PENDFREE
- PENDRECEIVE

- RECEIVE
- ROLLBACK
- SEND
- SYNCFREE
- SYNCRECEIVE
- SYNCSEND

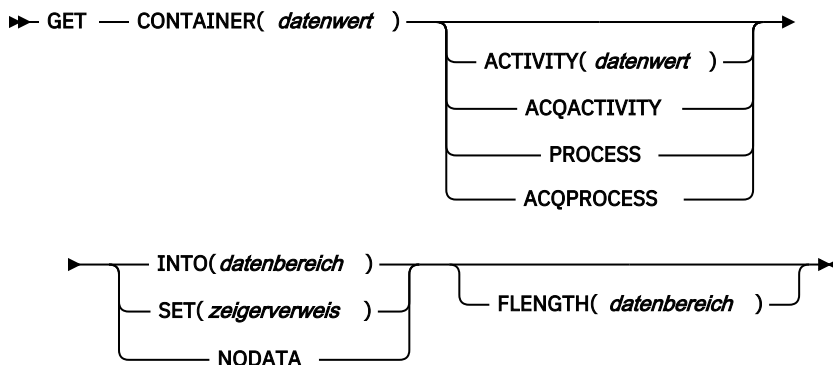
Tabelle 14. Rückgabecodes für den Befehl GDS WAIT

RETCODE (hexadezimal)	Beschreibung
03 00	CONVID ist für eine Sitzung, die keine APPC-Sitzung ist.
03 00	CONVID ist für eine Sitzung, die von CPI Communications verwendet wird.
03 04	CONVID ist für einen Dialog, der kein Basisdialog ist.
03 08	Eine Statusprüfung hat stattgefunden.
04	CONVID ist für eine Sitzung, die der Task nicht zugeordnet ist oder die ein Relay-Link ist.

GET CONTAINER (BTS)

Ruft Daten aus einem angegebenen BTS-Datencontainer ab.

GET CONTAINER



Bedingungen: ACTIVITYERR, CONTAINERERR, INVREQ, IOERR, LENGERR, LOCKED, PROCESSBUSY

Beschreibung

Der Befehl GET CONTAINER liest die Daten, die einer angegebenen BTS-Aktivität oder einem angegebenen BTS-Prozess zugeordnet sind, in den Arbeitsspeicher ein.

Der Container, der die Daten enthält, wird durch den Namen und durch den Prozess oder die Aktivität identifiziert, für den bzw. die er ein Container ist, d. h. der Prozess oder die Aktivität ist "Eigner" des Containers. Die Aktivität, die Eigner des Containers ist, kann wie folgt angegeben werden:

- Explizit durch Angabe einer der Optionen für Prozess oder Aktivität.
- Implizit durch Weglassen der Optionen für Prozess oder Aktivität. Wenn diese Optionen nicht angegeben werden, wird die aktuelle Aktivität impliziert.

Sie auch [PUT CONTAINER \(BTS\)](#) und [MOVE CONTAINER \(BTS\)](#).

Optionen

ACQACTIVITY

Gibt eine der folgenden Informationen an:

- Wenn das Programm, das den Befehl absetzt, einen Prozess angefordert hat, gibt diese Option an, dass der Eigner des Containers die Stammaktivität dieses Prozesses ist.
- Andernfalls gibt diese Option an, dass der Container der Aktivität gehört, die von dem Programm durch einen Befehl ACQUIRE ACTIVITYID angefordert wurde.

ACQPROCESS

Gibt an, dass der Container dem Prozess gehört, den das Programm, das den Befehl absetzt, in der aktuellen Arbeitseinheit angefordert hat.

ACTIVITY(datenwert)

Gibt den Namen (1 - 16 Zeichen) der Aktivität an, die Eigner des Containers ist. Dies muss eine untergeordnete Aktivität der aktuellen Aktivität sein.

CONTAINER(datenwert)

Gibt den Namen (1 - 16 Zeichen) des Containers an, der die abzurufenden Daten enthält.

FLENGTH(datenbereich)

Als Eingabefeld gibt FLENGTH die Länge der zu lesenden Daten als Vollwort-Binärwert an. Als Ausgabefeld gibt FLENGTH die Länge der Daten im Container zurück. Ob FLENGTH ein Eingabefeld oder ein Ausgabefeld ist, hängt davon ab, welche der Optionen INTO, SET oder NODATA Sie angeben.

Angabe der Option INTO

FLENGTH ist sowohl ein Eingabe- als auch ein Ausgabefeld.

Als **Eingabefeld** gibt FLENGTH die maximale Länge der Daten an, die das Programm akzeptiert. Wenn der angegebene Wert kleiner null ist, wird null angenommen. Wenn die Länge der Daten den angegebenen Wert überschreitet, werden die Daten auf diesen Wert abgeschnitten und die Bedingung LENGERR tritt auf. Ist die Länge der Daten kürzer als der angegebene Wert, werden die Daten ohne Auffüllung kopiert und die Bedingung LENGERR tritt auf.

Die Option FLENGTH muss nicht angegeben werden, wenn die Länge durch den Compiler aus der Variablen INTO generiert werden kann. Wenn Sie sowohl INTO als auch FLENGTH angeben, gibt FLENGTH die maximale Länge der Daten an, die das Programm akzeptiert.

Bei Verwendung als **Ausgabefeld** (d. h. nach Abschluss der Abrufoperation) setzt CICS den Datenbereich, sofern angegeben, auf die tatsächliche Länge der Daten in dem Container.

Angabe der Option SET oder NODATA

FLENGTH ist nur Ausgabefeld. Die Angabe ist erforderlich und sie muss einen Datenbereich angeben.

Nach Abschluss der Abrufoperation wird der Datenbereich auf die tatsächliche Länge der Daten im Container gesetzt.

INTO(datenbereich)

Gibt einen Bereich des Arbeitsspeichers an, in den die abzurufenden Daten eingelesen werden sollen.

NODATA

Gibt an, dass keine Daten abgerufen werden sollen. Verwenden Sie diese Option, um die Länge der Daten in dem Container (die in FLENGTH zurückgegeben wird) zu ermitteln.

PROCESS

Gibt an, dass der abzurufende Container dem aktuellen Prozess gehört, das heißt, dem Prozess, für den das Programm ausgeführt wird, das den Befehl absetzt.

SET(zeigerverweis)

Gibt einen Zeigerverweis an, in dem die Adresse der abgerufenen Daten zurückgegeben wird. Der Datenbereich wird von CICS beibehalten, bis ein nachfolgender Befehl GET CONTAINER mit der Option SET von der Task abgesetzt wird oder bis die Task beendet wird.

Wenn Ihre Anwendung die Daten behalten muss, sollte sie sie in einen eigenen Speicher verschieben.

Bedingungen

109 ACTIVITYERR

RESP2-Werte:

8

Die in der Option ACTIVITY angegebene Aktivität konnte nicht gefunden werden.

110 CONTAINERERR

RESP2-Werte:

10

Der in der Option CONTAINER angegebene Container konnte nicht gefunden werden.

16 INVREQ

RESP2-Werte:

2

Die Option INTOCCSID wurde ohne die Option CHANNEL angegeben und es ist kein aktueller Kanal vorhanden (weil an das Programm, das den Befehl abgesetzt hat, kein Kanal übergeben wurde). Die Option INTOCCSID ist nur für Befehle GET CONTAINER gültig, die einen Kanal (explizit oder implizit) angeben. Sie ist für Befehle GET CONTAINER (BTS) nicht gültig.

4

Der Befehl wurde außerhalb des Gültigkeitsbereichs einer momentan aktiven Aktivität abgesetzt.

15

Die Option ACQPROCESS wurde verwendet, jedoch hat die Arbeitseinheit, die die Anforderung abgesetzt hat, keinen Prozess angefordert.

24

Die Option ACQACTIVITY wurde verwendet, jedoch hat die Arbeitseinheit, die die Anforderung abgesetzt hat, keine Aktivität angefordert.

25

Die Option PROCESS wurde verwendet, jedoch wurde der Befehl außerhalb des Gültigkeitsbereichs eines momentan aktiven Prozesses abgesetzt.

17 IOERR

RESP2-Werte:

30

Es ist ein Ein-/Ausgabefehler für die Repository-Datei aufgetreten.

31

Der Datensatz in der Repository-Datei wird gerade verwendet.

22 LENGERR

RESP2-Werte:

11

Die Länge des Programmbereichs stimmt nicht mit der Länge der Daten in dem Container überein. Wenn der Bereich kleiner ist, werden die Daten abgeschnitten, um sie in den Bereich einzupassen. Ist der Bereich größer, werden die Daten in den Programmbereich kopiert, jedoch wird keine Auffüllung hinzugefügt.

100 LOCKED

Die Anforderung kann nicht ausgeführt werden, weil eine Sperre für den relevanten Datensatz in der Repository-Datei besteht.

106 PROCESSBUSY

RESP2-Werte:

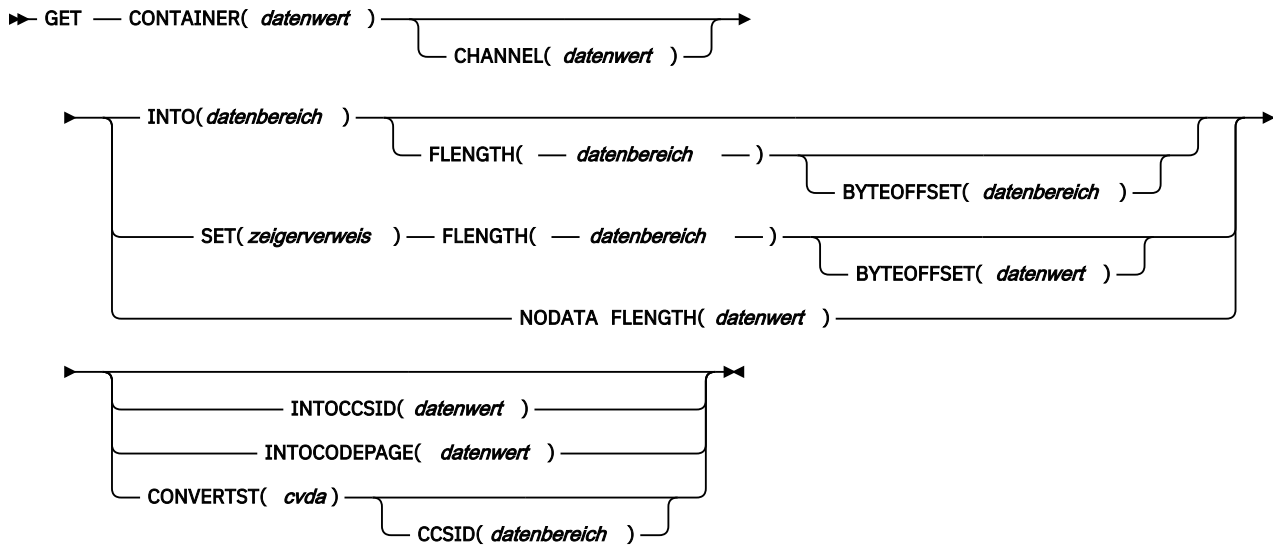
13

Die Anforderung konnte nicht erfüllt werden, weil der Prozessdatensatz durch eine andere Task gesperrt ist.

GET CONTAINER (CHANNEL)

Ruft Daten aus einem benannten Kanalcontainer ab.

GET CONTAINER (CHANNEL)



Bedingungen: CCSIDERR, CHANNELERR, CODEPAGEERR, CONTAINERERR, INVREQ, LENGERR

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl GET CONTAINER (CHANNEL) liest die Daten, die einem angegebenen Kanalcontainer zugeordnet sind.

Der Container, der die Daten enthält, wird durch den Namen und den Kanal identifiziert, für den er ein Container ist, d. h. den Kanal, der "Eigner" des Containers ist.

Der Kanal, der Eigner des Containers ist, kann explizit durch Angabe der Option CHANNEL oder implizit durch Weglassen der Option CHANNEL identifiziert werden. Wenn diese Option nicht angegeben wird, wird der aktuelle Kanal impliziert. Wenn kein aktueller Kanal vorhanden ist, wird der Befehl als Befehl **EXEC CICS GET CONTAINER** (BTS) interpretiert. Weitere Informationen finden Sie unter „[GET CONTAINER \(BTS\)](#)“ auf Seite 233.

Optionen

BYTEOFFSET(datenwert)

Gibt die relative Position (Offset) in Byte an, an der die zurückgegebenen Daten beginnen. Für CHAR-Container wird der Wert von BYTEOFFSET als relative Position in den Daten in der angeforderten Codepage verwendet. Wenn Sie eine Codepage mit Mehrbytezeichen verwenden, können die zurückgegebenen Daten abhängig von dem in BYTEOFFSET angegebenen Wert Teile von Zeichen am Anfang, am Ende oder an beiden Positionen enthalten. In diesem Fall muss Ihr Anwendungsprogramm in der Lage sein, die zurückgegebenen Daten zu verarbeiten und zu interpretieren. Wenn der angegebene Wert kleiner null ist, wird null angenommen.

CCSID(datenbereich)

Gibt ein Vollwort zurück, das die ID des codierten Zeichensatzes (Coded Character Set Identifier - CCSID) der Daten enthält, die durch die Option CONVERTST(NOCONVERT) zurückgegeben werden. Sie können diese Option dazu verwenden, Container mit dem Datentyp CHAR abzurufen, ohne die Daten zu konvertieren. Wenn der Datentyp BIT für den Container angegeben wurde, ist dieser Wert null.

CHANNEL(datenwert)

Gibt den Namen (1 - 16 Zeichen) des Kanals an, der Eigner des Containers ist. Sie können den Kanalnamen DFHTRANSACTION angeben, um den Transaktionskanal zu verwenden.

CONTAINER(datenwert)

Gibt den Namen (1 - 16 Zeichen) des Containers an, der die abzurufenden Daten enthält.

CONVERTST(cvda)

Gibt den erforderlichen Datenkonvertierungsstatus an.

NOCONVERT

Die Containerdaten werden ohne Konvertierung abgerufen. Wenn Sie den Befehl WEB RECEIVE verwendet haben, um den HTTP-Hauptteil in einem Container zu speichern, und den Hauptteil unkonvertiert aus diesem Container abrufen müssen, müssen Sie die Option NOCONVERT verwenden.

FLENGTH(datenbereich)

Als Eingabefeld gibt FLENGTH die Länge der zu lesenden Daten als Vollwort-Binärwert an. Als Ausgabefeld gibt FLENGTH die Länge der Daten im Container zurück. FLENGTH ist abhängig von den angegebenen Optionen BYTEOFFSET, INTO, SET oder NODATA ein Eingabefeld oder ein Ausgabefeld.

Angabe der Option BYTEOFFSET

FLENGTH ist sowohl ein Eingabe- als auch ein Ausgabefeld.

Als **Eingabefeld** gibt FLENGTH die maximale Länge der Daten an, die das Programm akzeptiert. Die zurückgegebenen Daten beginnen an der relativen Position, die durch den Wert in BYTEOFFSET angegeben wird. Falls der angegebene Wert kleiner als Null ist, wird Null angenommen.

Bei Verwendung als **Ausgabefeld** (d. h. nach Abschluss der Abrufoperation) setzt CICS den Datenbereich auf die Länge der zurückgegebenen Daten. Die maximale Länge der zurückgegebenen Daten ist gleich der Länge der Daten in dem Container minus des Werts von BYTEOFFSET.

Angabe der Option INTO

FLENGTH ist sowohl ein Eingabe- als auch ein Ausgabefeld.

Als **Eingabefeld** gibt FLENGTH die maximale Länge der Daten an, die das Programm akzeptiert. Wenn der angegebene Wert kleiner null ist, wird null angenommen. Wenn die Länge der Daten den angegebenen Wert überschreitet, werden die Daten auf diesen Wert abgeschnitten und die Bedingung LENGERR tritt auf. Ist die Länge der Daten kürzer als der angegebene Wert, werden die Daten kopiert, es erfolgt jedoch keine Auffüllung.

Sie müssen die Option FLENGTH nicht angeben, wenn die Länge durch den Compiler aus der Variablen INTO generiert werden kann. Wenn Sie sowohl INTO als auch FLENGTH angeben, gibt FLENGTH die maximale Länge der Daten an, die das Programm akzeptiert.

Bei Verwendung als **Ausgabefeld** (d. h. nach Abschluss der Abrufoperation) setzt CICS den Datenbereich, sofern angegeben, auf die tatsächliche Länge der Daten in dem Container. Wenn der Container Zeichendaten enthält, die von einer CCSID in eine andere konvertiert wurden, ist dies die Länge der Daten nach der Konvertierung.

Angabe der Option SET oder NODATA

FLENGTH ist nur Ausgabefeld. Es muss vorhanden sein und es muss als Datenbereich angegeben werden.

Nach Abschluss der Abrufoperation wird der Datenbereich auf die tatsächliche Länge der Daten im Container gesetzt. Wenn der Container Zeichendaten enthält, die von einer CCSID in eine andere konvertiert wurden, ist dies die Länge der Daten nach der Konvertierung.

INTO(datenbereich)

Gibt den Datenbereich an, in den die abgerufenen Daten eingefügt werden sollen.

INTOCCSID(datenwert)

Gibt die ID des codierten Zeichensatzes (Coded Character Set Identifier - CCSID), in den die Zeichendaten im Container konvertiert werden, als Vollwort-Binärwert an. Wenn Sie es bevorzugen, einen IANA-Namen für die Codepage anzugeben, oder wenn Sie es bevorzugen, die CCSID in Form alphanumerischer Zeichen anzugeben, verwenden Sie stattdessen die Option INTOCODEPAGE.

Für CICS Transaction Server for z/OS-Anwendungen ist die CCSID in der Regel eine EBCDIC-CCSID. Es ist jedoch möglich, eine ASCII-CCSID anzugeben, wenn Sie zum Beispiel ASCII-Daten abrufen wollen, ohne dass diese automatisch in EBCDIC konvertiert werden.

Wenn weder INTOCCSID noch INTOCODEPAGE angegeben werden, wird standardmäßig die CCSID der Region als Wert für die Konvertierung angenommen. Die Standard-CCSID der Region wird im Systeminitialisierungsparameter **LOCALCCSID** angegeben.

Es können nur Zeichendaten konvertiert werden und nur dann, wenn der Datentyp CHAR im Befehl **PUT CONTAINER** oder **PUT64 CONTAINER** angegeben wurde, der zum Einfügen der Daten in den Container verwendet wurde. Der Datentyp CHAR wird impliziert, wenn die Option FROMCCSID oder FROMCODEPAGE im Befehl **PUT CONTAINER** oder **PUT64 CONTAINER** angegeben wurde.

Weitere Informationen zur Datenkonvertierung mit Kanälen finden Sie unter [Datenkonvertierung mit Kanälen](#).

Eine Erläuterung von CCSIDs finden Sie unter [Codepagekonvertierung mit Kanälen vorbereiten](#).

INTOCODEPAGE(datenwert)

Gibt einen IANA-registrierten alphanumerischen Zeichensatznamen oder eine ID des codierten Zeichensatzes (CCSID) für die Codepage an, in die die Zeichendaten in dem Container konvertiert werden sollen, wobei bis zu 40 alphanumerische Zeichen mit geeigneter Interpunktion verwendet werden. Verwenden Sie diese Option anstelle der Option CCSID, wenn Sie einen von IANA registrierten Zeichensatznamen verwenden wollen, was im Header 'Content-Type' für eine HTTP-Anforderung angegeben wird. CICS konvertiert den IANA-Namen in eine CCSID. Der nachfolgende Datenkonvertierungsprozess ist identisch. Verwenden Sie diese Option außerdem dann, wenn Sie es bevorzugen, die CCSID in alphanumerischen Zeichen und nicht als Vollwort-Binärwert anzugeben.

Wenn ein IANA-Name für eine Codepage vorhanden ist und CICS die Verwendung der Codepage unterstützt, wird der Name mit der CCSID aufgelistet. Weitere Informationen finden Sie unter [Codepagekonvertierung mit Kanälen vorbereiten](#).

NODATA

Gibt an, dass keine Daten abgerufen werden. Verwenden Sie diese Option, um die Länge der Daten in dem Container (die in FLENGTH zurückgegeben wird) zu ermitteln.

Die Länge der Zeichendaten kann sich bei einer Datenkonvertierung ändern. Wenn Zeichendaten also in eine andere CCSID *als die dieser Region* konvertiert werden sollen, sollten Sie, wenn Sie NODATA angeben, auch die Option INTOCCSID angeben. Dadurch wird sichergestellt, dass die richtige Länge der konvertierten Daten in FLENGTH zurückgegeben wird.

SET(zeigerverweis)

Gibt einen Zeigerverweis an, in dem die Adresse der abgerufenen Daten zurückgegeben wird.

Wenn das Anwendungsprogramm, das den Befehl **GET CONTAINER** absetzt, mit DATALOCATION(ANY) definiert ist, kann die Adresse der Daten oberhalb oder unterhalb der 16-MB-Grenze liegen. Wenn das Anwendungsprogramm mit DATALOCATION(BELOW) definiert ist, liegt die Adresse der Daten unterhalb der 16-MB-Grenze. Wenn TASKDATAKEY(USER) für die ausführende Transaktion angegeben ist, werden die Daten im Benutzerschlüssel zurückgegeben. Andernfalls befinden sie sich im CICS-Schlüssel.

CICS verwaltet den Datenbereich, bis eines der folgenden Ereignisse auftritt:

- Ein nachfolgender Befehl **GET CONTAINER** oder **GET64 CONTAINER** mit der Option SET für denselben Container im selben Kanal wird durch ein Programm abgesetzt, das auf diesen Speicher zugreifen kann.
- Der Container wird durch einen Befehl **DELETE CONTAINER** gelöscht.
- Der Container wird durch einen Befehl **MOVE CONTAINER** verschoben.
- Der Kanal verlässt den Bereich des Programms.
- Der Kanal und die Container, die sich darin befinden, werden durch einen Befehl **DELETE CHANNEL** gelöscht.

Beachten Sie Verbindungen zu anderen Programmen, die einen dieser Befehle absetzen können.

Setzen Sie keinen Befehl **FREEMAIN** ab, um diesen Speicher freizugeben.

Wenn Ihre Anwendung die Daten behalten muss, sollte sie sie in einen eigenen Speicher verschieben.

Bedingungen

123 CCSIDERR

RESP2-Werte:

- 1**
Die in der Option INTOCCSID angegebene CCSID liegt außerhalb des Bereichs gültiger CCSID-Werte.
- 2**
Die in der Option INTOCCSID angegebene CCSID und die CCSID des Containers sind eine nicht unterstützte Kombination. (Die CCSID des Containers ist der Wert, der bei der Erstellung des Containers in der Option FROMCODEPAGE oder FROMCCSID angegeben oder als Standardwert angenommen wurde.)
- 3**
Die Daten wurden mit dem Datentyp BIT erstellt. Eine Codepagekonvertierung ist nicht möglich. Die Daten wurden ohne Codepagekonvertierung zurückgegeben.
- 4**
Mindestens ein Zeichen konnte nicht konvertiert werden. Das Zeichen wurde durch ein Leerzeichen in den konvertierten Daten ersetzt.
- 5**
Bei der Codepagekonvertierung eines Containers trat ein interner Fehler auf.

122 CHANNELERR

RESP2-Werte:

- 2**
Der in der Option CHANNEL angegebene Kanal konnte nicht gefunden werden.

125 CODEPAGEERR

RESP2-Werte:

- 1**
Die in der Option INTOCODEPAGE angegebene Codepage wird nicht unterstützt.
- 2**
Die in der Option INTOCODEPAGE angegebene Codepage und die Codepage des Kanals sind eine nicht unterstützte Kombination.
- 3**
Die Daten wurden mit dem Datentyp BIT erstellt. Eine Codepagekonvertierung ist nicht möglich. Die Daten wurden ohne Codepagekonvertierung zurückgegeben.
- 4**
Mindestens ein Zeichen konnte nicht konvertiert werden. Das Zeichen wurde durch ein Leerzeichen in den konvertierten Daten ersetzt.
- 5**
Bei der Codepagekonvertierung eines Containers trat ein interner Fehler auf.

110 CONTAINERERR

RESP2-Werte:

- 10**
Der in der Option CONTAINER angegebene Container konnte nicht gefunden werden.

16 INVREQ

RESP2-Werte:

- 2**
Die Option INTOCCSID wurde ohne die Option CHANNEL angegeben und es ist kein aktueller Kanal vorhanden (weil an das Programm, das den Befehl abgesetzt hat, kein Kanal übergeben wurde). Die Option INTOCCSID ist nur für Befehle GET CONTAINER gültig, die einen Kanal (explizit oder implizit) angeben.

4

Die Option CHANNEL wurde nicht angegeben, es ist kein aktueller Kanal vorhanden (weil an das Programm, das den Befehl abgesetzt hat, kein Kanal übergeben wurde) und der Befehl wurde außerhalb des Gültigkeitsbereichs der momentan aktiven BTS-Aktivität abgesetzt.

5

Der CVDA-Wert für CONVERTST ist ungültig.

22 LENGERR

RESP2-Werte:

11

Die Länge des Programmbereichs ist kürzer als die Länge der Daten in dem Container. Wenn der Bereich kleiner ist, werden die Daten abgeschnitten, um sie in den Bereich einzupassen.

12

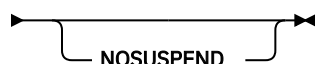
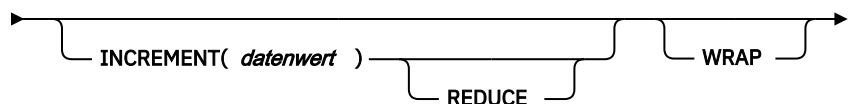
Die relative Position (Offset) ist größer oder gleich der Länge des Containers.

GET COUNTER und GET DCOUNTER

Ruft die nächste Nummer aus dem benannten Zähler im angegebenen Pool ab. COUNTER wird für Vollwortsähler mit Vorzeichen verwendet, DCOUNTER für Doppelwortsähler ohne Vorzeichen.

GET COUNTER

➔ GET — COUNTER(*name*) — POOL(*name*) — VALUE(*datenbereich*) —➔

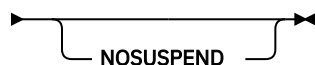
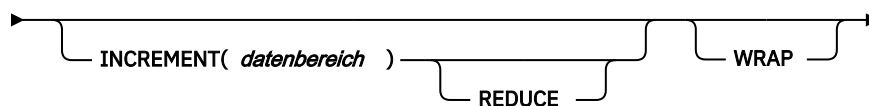


Bedingungen: BUSY, INVREQ, LENGERR, SUPPRESSED

Dieser Befehl ist threadsicher.

GET DCOUNTER

➔ GET — DCOUNTER(*name*) — POOL(*name*) — VALUE(*datenbereich*) —➔



Bedingungen: BUSY, INVREQ, LENGERR, SUPPRESSED

Dieser Befehl ist threadsicher.

Beschreibung

Diese Zählerbefehle rufen vom Server für benannte Zähler die aktuelle Nummer aus dem benannten Zähler im angegebenen Pool ab und aktualisieren die aktuelle Nummer um das Standardinkrement oder um ein angegebenes Inkrement. Das Standardinkrement ist 1.

Mithilfe der Optionen COMPAREMAX und COMPAREMIN können Sie eine Nummer nur abrufen, wenn sie in einen angegebenen Bereich fällt bzw. wenn sie oberhalb oder unterhalb eines angegebenen Werts liegt.

Informationen zur Angabe von Vollwort- und Doppelwortvariablen bei diesen Befehlen für benannte Zähler finden Sie unter „[Argumentwerte für CICS-Befehle](#)“ auf Seite 2.

Optionen

COMPAREMAX(*datenwert*)

Gibt einen Vollwort-Binärwert mit Vorzeichen (bzw. einen Doppelwort-Binärwert ohne Vorzeichen für DCOUNTER) für einen Wert an, der mit dem aktuellen Wert des benannten Zählers verglichen werden soll, und macht das Ergebnis des Befehls GET von dem Vergleich abhängig:

- Wenn der aktuelle zuzuweisende Wert kleiner oder gleich dem Wert ist, der im Parameter COMPAREMAX angegeben wird, wird der aktuelle Wert zurückgegeben und die Antwort ist normal.
- Wenn der aktuelle Wert größer als der angegebene Wert ist, gibt CICS eine Ausnahmebedingung zurück.

Normalerweise ist der Wert von COMPAREMAX größer als der Wert von COMPAREMIN und der aktuelle Wert muss beide Vergleichskriterien erfüllen (d. h., er muss zwischen den beiden Werten liegen oder mit einem der Werte identisch sein).

Sie können einen Wert für COMPAREMAX angeben, der kleiner als der Wert für COMPAREMIN ist. In dieser Situation wird für den aktuellen Wert angenommen, dass er im gültigen Bereich liegt, wenn er entweder den mit COMPAREMIN oder den mit COMPAREMAX angegebenen Vergleich erfüllt.

COMPAREMIN(*datenwert*)

Gibt einen Vollwort-Binärwert mit Vorzeichen (bzw. einen Doppelwort-Binärwert ohne Vorzeichen für DCOUNTER) für einen Wert an, der mit dem aktuellen Wert des benannten Zählers verglichen werden soll, und macht das Ergebnis des Befehls GET von dem Vergleich abhängig:

- Wenn der aktuelle zuzuweisende Wert gleich oder größer als der Wert ist, der im Parameter COMPAREMIN angegeben wird, wird der aktuelle Wert zurückgegeben und die Antwort ist normal.
- Wenn der aktuelle Wert kleiner als der angegebene Wert ist, gibt CICS eine Ausnahmebedingung zurück.

Anmerkung: Sie können einen Wert für COMPAREMIN angeben, der größer als der Wert für COMPAREMAX ist. Informationen zur Auswirkung einer solchen Angabe finden Sie unter dem Parameter COMPAREMAX.

COUNTER(*name*)

Gibt den Namen des Vollwort-Zählers an, aus dem die aktuelle Nummer dem Anwendungsprogramm zugewiesen werden soll. Der Name kann bis zu 16 alphanumerische Zeichen lang sein. Falls es sich bei *name* um eine Variable handelt, die einen weniger als 16 Zeichen langen Namen enthält, muss der Name mit nachgestellten Leerzeichen aufgefüllt werden.

DCOUNTER(*name*)

Gibt den Namen des Doppelwort-Zählers an, aus dem die aktuelle Nummer dem Anwendungsprogramm zugewiesen werden soll. Der Name kann bis zu 16 alphanumerische Zeichen lang sein. Falls es sich bei *name* um eine Variable handelt, die einen weniger als 16 Zeichen langen Namen enthält, muss der Name mit nachgestellten Leerzeichen aufgefüllt werden.

INCREMENT(*datenwert*)

Gibt anstelle des Standardwerts 1 einen Vollwort-Binärwert mit Vorzeichen (bzw. einen Doppelwort-Binärwert ohne Vorzeichen für DCOUNTER) für ein Inkrement an, um das der benannte Zähler aktualisiert werden soll. Der Zähler wird um das Inkrement erhöht, nachdem die aktuelle Nummer zugewiesen wurde.

Die Angabe eines Inkrements zum Überschreiben des Standardinkrements 1 ermöglicht es dem Anwendungsprogramm, die exklusive Nutzung von mehr als einer Nummer für jeden Aufruf anzufordern. Wenn zum Beispiel die exklusive Nutzung eines Blocks von 20 Nummern angefordert werden soll, geben Sie INCREMENT(20) an.

In der Beschreibung der Optionen REDUCE und WRAP finden Sie Information zur Auswirkung der Angabe eines Inkrements, wenn sich der Zähler am oder nahe am Maximalwert befindet.

NOSUSPEND

Gibt eine sofortige Rückkehr zum Anwendungsprogramm während einer Neuerstellung der Coupling-Facility-Struktur an. Wenn Sie den Parameter NOSUSPEND nicht angeben, wird die Anforderung so lange wiederholt, bis sie erfolgreich ist. Wenn Sie NOSUSPEND angeben, schlägt die Anforderung sofort fehl und CICS gibt die Bedingung BUSY, RESP2=500 zurück.

POOL(poolname)

Gibt eine acht Zeichen lange Zeichenfolge an, die als Poolauswahlparameter verwendet werden soll, um den Pool auszuwählen, in dem sich der benannte Zähler befindet. Die Zeichenfolge kann ein logischer Poolname oder der tatsächliche Poolname sein.

Gültige Zeichen für die Poolselektorenzeichenfolge sind A bis Z, 0 bis 9, \$ @ # und _ (Unterstrichzeichen). Falls es sich bei *name* um eine Variable handelt, die einen weniger als 8 Zeichen langen Namen enthält, muss der Name mit nachgestellten Leerzeichen aufgefüllt werden.

Dieser Parameter ist optional. Wenn Sie den Namen des Pools nicht angeben, wird ein Poolauswahlwert aus 8 Leerzeichen angenommen.

Wenn kein entsprechender Eintrag in der Optionstabelle DFHNCOPT vorhanden ist, verwendet CICS den Namen, der im Systeminitialisierungsparameter NCPLDFT angegeben ist, der den Standardpool für benannte Zähler angibt.

Informationen zum Generieren einer Optionstabelle für benannte Zähler mithilfe des Makros DFHNCO finden Sie unter [Server für benannte Zähler](#).

REDUCE

Gibt an, dass der Server für benannte Zähler das angegebene Inkrement verringern soll, wenn der verbleibende Bereich der zuzuweisenden Nummern zu klein ist.

Der Bereich der Nummern ist zu klein, wenn die Differenz zwischen dem aktuellen Wert und dem Maximalwert plus 1 kleiner als das angegebene Inkrement ist. In diesem Fall gilt Folgendes:

- Wenn Sie REDUCE angeben, wird der Wert des Parameters INCREMENT verringert und die GET-Anforderung wird erfolgreich ausgeführt. In diesem Fall hat der Befehl GET einen Bereich von Nummern reserviert, der kleiner als der durch den Parameter INCREMENT angegebene Bereich ist, und der aktuelle Wert wird auf den Maximalwert plus 1 aktualisiert.
- Wenn Sie die Option REDUCE nicht angeben, hängt das Ergebnis davon ab, ob Sie die Option WRAP angeben. Wenn weder die Option REDUCE noch die Option WRAP angegeben werden, schlägt die Anforderung mit einem Fehler 'Zähler am Grenzwert' (SUPPRESSED, RESP2=101) fehl, jedoch wird die aktuelle Nummer nicht geändert. Beispiel: Wenn eine Anforderung für den Parameter INCREMENT den Wert 15 angibt, die aktuelle Nummer 199 990 ist und die maximale Nummer mit 199 999 definiert ist, schlägt der Befehl GET fehl, weil eine Aktualisierung des Zählers um das angegebene Inkrement dazu führen würde, dass die aktuelle Nummer über dem Wert 200 000 läge.

VALUE(datenbereich)

Gibt den Datenbereich (Vollwort-Datenbereich mit Vorzeichen für COUNTER und Doppelwort-Datenbereich ohne Vorzeichen für DCOUNTER) an, in den CICS die aktuelle Nummer zurückgibt, die vom Server für benannte Zähler für den angegebenen Pool abgerufen wurde.

WRAP

Gibt an, dass der Server für benannte Zähler den benannten Zähler automatisch zurücksetzen soll, wenn er sich in einer Bedingung 'Zähler am Grenzwert' befindet, um so die ansonsten auftretende Fehlerbedingung zu vermeiden.

Wenn sich der benannte Zähler in der Bedingung 'Zähler am Grenzwert' befindet oder wenn das Inkrement, das ohne die Option REDUCE angegeben wurde, die Bedingung 'Zähler am Grenzwert' zur Folge hätte, reagiert der Zählerserver wie folgt:

- Er setzt den aktuellen Wert des benannten Zählers auf den Minimalwert zurück, der für den Zähler definiert ist.
- Er gibt einen neuen aktuellen Wert an das Anwendungsprogramm mit DFHRESP(NORMAL) zurück.
- Er aktualisiert den aktuellen Wert um das erforderliche Inkrement, sodass er für die nächste Anforderung bereit ist.

Wenn Sie die Option WRAP nicht angeben und die Bedingung 'Zähler am Grenzwert' erreicht wird, gibt CICS die Bedingung SUPPRESSED, RESP2=101 zurück.

Bedingungen

128 BUSY

RESP2-Werte:

500

Die Option NOSUSPEND wurde im Befehl angegeben und die Coupling-Facility-Struktur ist momentan während einer Neuerstellung nicht verfügbar.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

RESP2-Werte:

201

Der benannte Zähler wurde nicht gefunden.

301

Der Server hat einen Fehlercode gemeldet, der für die Schnittstelle des benannten Zählers unverständlich ist. Dies ist in der Regel nur dann möglich, wenn das Schnittstellenlademodul DFHNCIF eine geringere Wartungsstufe oder ein niedrigeres Release-Level als der Server selbst aufweist.

303

Bei einem Makro, das für den Zugriff auf die Coupling-Facility verwendet wird, trat ein unerwarteter Fehler wie beispielsweise eine Strukturstörung oder eine Verbindungsunterbrechung auf. Weitere Informationen enthält die Nachricht DFHNC0441 im Jobprotokoll der Anwendung.

304

Der im Programm angegebene Parameter für die Poolauswahl kann anhand der aktuellen Optionstabelle nicht in einen gültigen Servernamen aufgelöst werden.

305

Die Schnittstelle kann keine Verbindung zum Server für den ausgewählten Pool mit benannten Zählern herstellen. Weitere Informationen enthält eine AXM-Servicenachricht (AXMSCnnnn) im Jobprotokoll der Anwendung.

306

Während der Server eine Anforderung verarbeitete, trat eine abnormale Beendigung auf. Weitere Informationen enthält eine Nachricht im Jobprotokoll der Anwendung und im Jobprotokoll des Servers.

308

Das Optionstabellenmodul DFHNCOPT, das zum Auflösen eines Poolnamens benötigt wird, konnte nicht geladen werden.

309

Während der Verarbeitung der Optionstabelle hat die Schnittstelle für benannte Zähler ein unbekanntes Eintragsformat festgestellt. Entweder wurde die Optionstabelle nicht ordnungsgemäß generiert oder das Schnittstellenlademodul DFHNCIF hat nicht dasselbe Release-Level wie die Optionstabelle.

310

Ein Optionstabelleneintrag, der mit dem angegebenen Poolnamen übereinstimmt, hat ein Benutzerexitprogramm angegeben, aber das Benutzerexitprogramm ist nicht durch eine Linkbearbeitung mit der Optionstabelle verbunden und kann nicht geladen werden.

311

Eine Antwort vom Server des benannten Zählers an das Schnittstellenmodul DFHNCIF der Client-region gibt an, dass gegenwärtig ein systemverwalteter erneuter Build ausgeführt wird, aber die Bedingung wird von der EXEC CICS-Schnittstelle nicht erkannt. Dies bedeutet, dass die CICS-Region den Stand von CICS TS 2.1 oder früher aufweist.

403

Der Parameter POOL enthält ungültige Zeichen oder eingebettete Leerzeichen.

404

Der Parameter COUNTER enthält ungültige Zeichen oder eingebettete Leerzeichen.

406

Der Wert für INCREMENT ist ungültig. Der angegebene Wert kann nicht größer als der Gesamtbereich des Zählers ((Maximalwert - Minimalwert) + 1) sein.

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

Die Bedingung LENGERR tritt nur für COUNTER-Befehle auf und gilt nicht für DCOUNTER-Befehle. Sie tritt auf, wenn ein Zähler, der durch einen DCOUNTER-Befehl oder durch die CALL-Schnittstelle definiert wurde, einen Wert enthält, der zu groß ist, um als Vollwort-Binärwert ohne Vorzeichen korrekt dargestellt zu werden (d. h., der Zähler verwendet mehr als 31 Bit).

In jedem der drei Fälle von Überlauf schließt der Server des benannten Zählers die Operation ab und gibt eine Warnungsantwort an CICS zurück, die CICS an Ihr Anwendungsprogramm als RESP2-Wert zurückgibt. Der Datenbereich enthält die niedrigstwertigen 32 Bit, die vom Server des benannten Zählers zurückgegeben werden, die eine negative Zahl darstellen könnten.

RESP2-Werte:

001

Der aktuelle Wert, den der Server versucht hat, im Datenbereich VALUE zurückzugeben, ist in ein höchstwertiges Bit (Vorzeichenbit) übergelaufen (d. h., der Wert ist negativ).

002

Der aktuelle Wert ist nur 1 Bit zu groß für einen Vollwort-Datenbereich. In diesem Fall ist der Überlaufwert genau 1.

003

Der aktuelle Wert ist um einen Wert größer 1 zu groß für einen Vollwort-Datenbereich.

Standardaktion: Die Task wird abnormal beendet.

72 SUPPRESSED

RESP2-Werte:

101

Der Maximalwert für den benannten Zähler wurde bereits zugewiesen und der Zähler befindet sich in der Bedingung 'Zähler am Grenzwert'. Es können keine weiteren Zählernummern mehr zugewiesen werden, bis der benannte Zähler zurückgesetzt wurde. Dies geschieht entweder durch einen Befehl REWIND oder durch Angabe der Option WRAP im Befehl GET.

103

Diese Bedingung tritt in einer der folgenden Situationen auf:

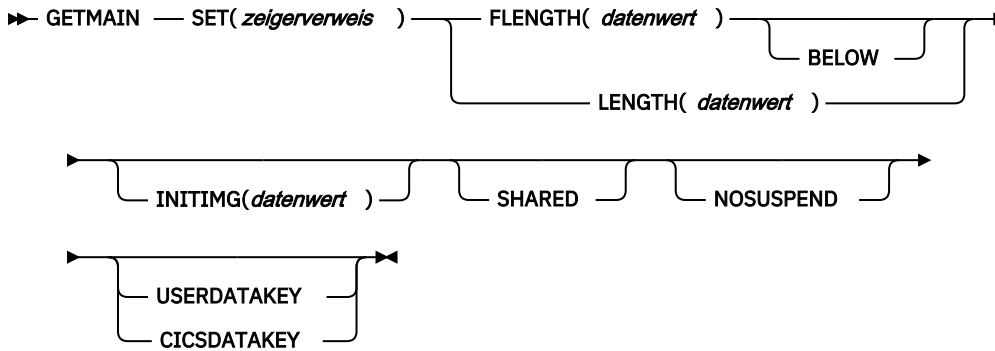
- Der aktuelle Wert des benannten Zählers liegt nicht in dem Bereich, der durch die Parameter **COMPAREMAX** und **COMPAREMIN** angegeben wird, wenn beide Parameter angegeben wurden.
- Der aktuelle Wert des benannten Zählers ist größer als der Parameter **COMPAREMAX** oder kleiner als der Parameter **COMPAREMIN**, wenn nur eine dieser Optionen angegeben wurde.

Standardaktion: Die Task wird abnormal beendet.

GETMAIN

Ruft den 24-Bit- oder 31-Bit-Hauptspeicher ab.

GETMAIN



Bedingungen: LENGERR, NOSTG

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl **GETMAIN** ruft einen Hauptspeicherbereich der Größe ab, die durch die Option FLENGTH angegeben wird. Die Adresse des Bereichs wird in dem Zeigerverweis zurückgegeben, der in der Option SET angegeben wird.

Dieser Befehl schließt die Option LENGTH nur aus Gründen der Kompatibilität ein. Programme, die mit aktuellen Versionen von CICS ausgeführt werden, verwenden die Option FLENGTH.

CICS ordnet Speicher immer an 16-Byte-Grenzen zu und rundet die angeforderte Länge bis zum nächsten 16-Byte-Vielfachen auf. Es gibt keine Standardinitialisierung, sodass Sie die Option INITIMG verwenden müssen, wenn Sie Speicher benötigen, der mit einer bestimmten Bit-Konfiguration initialisiert wird.

CICS ordnet Speicher aus einem der folgenden dynamischen Speicherbereiche (DSAs) zu:

- DSAs im 24-Bit-Speicher:
 - Der dynamische CICS-Speicherbereich (CDSA) unterhalb von 16 MB (unterhalb der 16-MB-Grenze)
 - Der dynamische Benutzerspeicherbereich (UDSA) unterhalb von 16 MB
 - Der gemeinsam genutzte dynamische Speicherbereich (SDSA) unterhalb 16 MB
- DSAs im 31-Bit-Speicher, dem erweiterten dynamischen Speicherbereich (EDSA):
 - Der erweiterte dynamische CICS-Speicherbereich (ECDSA) oberhalb von 16 MB, jedoch unterhalb von 2 GB (oberhalb der 16-MB-Grenze)
 - Der erweiterte dynamische Benutzerspeicher (EUDSA) oberhalb von 16 MB, jedoch unterhalb von 2 GB
 - Der erweiterte gemeinsam genutzte dynamische Speicherbereich (ESDSA) oberhalb von 16 MB, jedoch unterhalb von 2 GB

Weitere Informationen zu diesen DSAs finden Sie unter [Dynamische CICS-Speicherbereiche](#).

Anmerkung: Der Befehl **GETMAIN** kann nicht zum Abrufen von Speicher aus den folgenden DSAs verwendet werden:

- Der schreibgeschützte DSA (RDSA).
- Der erweiterte schreibgeschützte DSA (ERDSA).
- Der erweiterte vertrauenswürdige DSA (ETDSA).
- DSAs im dynamischen Speicherbereich oberhalb der 2-GB-Grenze (GDSA). Zum Abrufen von Speicher aus diesen DSAs verwenden Sie den Befehl [GETMAIN64](#).

CICS ordnet Speicher aus dem 24- oder 31-Bit-Speicher abhängig von den folgenden Optionen zu:

- Die Option **FLENGTH** mit gleichzeitiger Angabe von **BELOW**. CICS ruft den Speicher aus einem DSA im 24-Bit-Speicher ab.
- Die Option **FLENGTH** ohne Angabe von **BELOW**. Der Adressierungsmodus (**AMODE**) des anfordernden Programms bestimmt die Position des abgerufenen Speichers. Beispiel: Für ein Programm mit **AMODE(31)** ruft CICS den Speicher aus dem **EDSA** ab.
- Die Option **LENGTH**. CICS ruft den Speicher aus einem DSA im 24-Bit-Speicher ab.

CICS ordnet Speicher aus einem CICS-Schlüssel oder aus dem gemeinsam genutzten DSA abhängig von den folgenden Optionen zu:

- **USERDATAKEY**
- **CICSDATAKEY**
- **SHARED**
- Wenn keine Datenschlüsseloption im Befehl **GETMAIN** angegeben wird, wird die Option **TASKDATAKEY** in der **TRANSACTION**-Ressourcendefinition verwendet, unter der das anfordernde Programm ausgeführt wird.

Siehe [TRANSACTION-Attribute](#).

Die Auswirkungen der Datenschlüsseloptionen werden in der folgenden Tabelle zusammengefasst:

Tabelle 15. Datenschlüsseloptionen im Befehl GETMAIN		
Ohne Datenschlüsseloption	Angabe von USERDATAKEY	Angabe von CICSDATAKEY
Speicherschlüssel wird durch TASKDATAKEY in der Transaktionsdefinition bestimmt.	Benutzerschlüsselspeicher. Speicher wird ohne Angabe der Option SHARED aus dem UDSA oder EUDSA und bei Angabe der Option SHARED aus dem SDSA oder ESDSA abgerufen.	CICS-Schlüsselspeicher. Speicher wird aus dem CDSA oder ECDSA abgerufen.

Die Datenschlüsseloption im Befehl **GETMAIN** überschreibt die Option **TASKDATAKEY** in der **TRANSACTION**-Ressourcendefinition. Sie können zum Beispiel die Option **CICSDATAKEY** angeben, um sicherzustellen, dass das anfordernde Programm CICS-Schlüsselspeicher aus einem CICS-DSA abrufen, auch wenn die Option **TASKDATAKEY(USER)** in der **TRANSACTION**-Ressourcendefinition angegeben ist.

Der Speicher, den eine Task erhält, ist verfügbar, bis er durch einen Befehl **FREEMAIN** oder **FREEMAIN64** freigegeben wird. Ein Bereich, der ohne die Option **SHARED** abgerufen wurde, kann nur durch die Task, die den Speicher abgerufen hat, freigegeben werden. Am Ende einer Task gibt CICS solchen Speicher automatisch frei, falls er noch nicht freigegeben wurde.

Jeder Speicher, der mit der Option **SHARED** abgerufen wird, ist für alle Tasks zugänglich, einschließlich der Tasks, die mit Transaktionsisolation ausgeführt werden. Allerdings wird ein Bereich mit der Option **SHARED** am Ende einer Task nicht freigegeben, sondern verbleibt, bis er explizit freigegeben wird. Jede Task kann die Anforderung **FREEMAIN** bzw. **FREEMAIN64** absetzen. Dies bedeutet, dass Sie **SHARED**-Speicher zur Kommunikation von einer Task zur anderen verwenden können.

Anmerkung: Die Verwendung des Befehls **GETMAIN** mit der Option **SHARED** kann möglicherweise transaktionsübergreifende Affinitäten erstellen, die sich negativ auf die Verwendung des dynamischen Transaktionsroutings auswirken. Weitere Informationen zu Transaktionsaffinitäten finden Sie unter [Affinität](#).

Optionen

BELOW

Gibt an, dass 24-Bit-Speicher (unterhalb der 16 MB-Grenze) abgerufen wird, das heißt, aus dem **CDSA**, **UDSA** oder **SDSA**.

CICSDATAKEY

Gibt an, dass CICS Speicher aus einem dynamischen CICS-Schlüsselspeicherbereich (CDSA oder ECDSA) zuordnet, wobei die in der Transaktionsressourcendefinition angegebene Option TASKDATAKEY überschrieben wird. Wenn Sie keine Datenschlüsseloption angeben, hängt der Speicherschlüssel (CICS-Schlüssel oder Benutzerschlüssel) von der Option TASKDATAKEY in der Transaktionsressourcendefinition ab.

Anmerkung: Wenn das Programm unter einer Task ausgeführt wird, die mit TASKDATAKEY(USER) in der Transaktionsressourcendefinition definiert ist, verwenden Sie den Befehl **FREEMAIN** oder **FREE-MAIN64** nicht explizit, sondern lassen Sie den Speicher im Rahmen der Taskbeendigung freigeben.

FLENGTH(datenwert)

Gibt die Anzahl Byte des erforderlichen Speichers im Vollwort-Binärformat an.

Die maximale Länge, die Sie angeben können, ist der Wert der Begrenzung für den entsprechenden dynamischen Speicherbereich (DSA), das heißt **DSALIMIT** oder **EDSALIMIT**. Dies sind die Systeminitialisierungsparameter, die die Gesamtspeicherbegrenzung definieren, innerhalb deren CICS Speicher zuordnen und die einzelnen DSAs im 24- bzw. 31-Bit-Speicher verwalten kann.

Wenn die angeforderte Länge größer als der entsprechende Wert von **DSALIMIT** oder **EDSALIMIT** ist, tritt die Bedingung LENGERR auf. Wenn die angeforderte Länge kleiner als die entsprechende Begrenzung, jedoch größer als der verfügbare Speicher ist, tritt die Bedingung NOSTG auf.

INITIMG(datenwert)

Gibt einen optionalen 1 Byte großen Initialisierungswert an. Wenn Sie INITIMG angeben, setzt CICS jedes Byte des angeforderten Speichers auf die Bitfolge, die Sie angeben. Andernfalls initialisiert CICS den Speicher nicht. Nur für COBOL-Programme gilt, dass Sie einen Datenbereich und keinen Datenwert verwenden müssen, um die Bitfolge für die Initialisierung zu definieren.

LENGTH(datenwert)

Diese Option wird nur aus Gründen der Kompatibilität mit Programmen unterstützt, die zur Ausführung unter früheren Releases von CICS geschrieben wurden. Es wird empfohlen, die Option FLENGTH zu verwenden.

Die Option LENGTH gibt die Anzahl Byte (Halbwort-Binärwert ohne Vorzeichen) des erforderlichen Speichers an. Diese Option impliziert Speicher von unterhalb der 16-MB-Grenze und hat eine Obergrenze von 65520 Byte. Für größere Bereiche oder Speicher oberhalb von 16 MB verwenden Sie die Option FLENGTH.

Wenn der Wert von LENGTH null ist, tritt eine Bedingung LENGERR auf. Wenn die angeforderte Länge größer als der verfügbare Speicher ist, tritt die Bedingung NOSTG auf.

NOSUSPEND

Gibt an, dass CICS, wenn kein Speicher verfügbar ist, die Task nicht aussetzt, sondern die Bedingung NOSTG ausgibt.

Wenn ein Befehl HANDLE CONDITION für NOSTG aktiv ist, wenn der Befehl ausgeführt wird, wird die Steuerung an die Benutzerbezeichnung übergeben, die in HANDLE CONDITION angegeben wurde. Dieses Verhalten hat Vorrang vor der Option NOSUSPEND, wird jedoch durch NOHANDLE oder RESP inaktiviert.

SET(zeigerverweis)

Setzt den Zeigerverweis auf die Adresse des angeforderten Hauptspeichers. Der Zeiger wird auf das erste Byte des Speicherbereichs gesetzt.

Durch eine Bedingung LENGERR, die auftritt, weil null oder ein negativer Wert für die Option FLENGTH oder LENGTH angegeben wurde, wird *zeigerverweis* auf null gesetzt.

SHARED

Verhindert die automatische Freigabe von Speicher, der durch einen Befehl **GETMAIN** abgerufen wurde, am Ende der Task, die ihn angefordert hat. Dies ermöglicht eine Task-zu-Task-Kommunikation. Ein Bereich, der mit der Option SHARED abgerufen wurde, wird erst freigegeben, wenn ein entsprechender Befehl **FREEMAIN** entweder durch die Task, die den Speicher angefordert hat, oder durch eine andere Task abgesetzt wird.

Beachten Sie, dass bei einer abnormalen Beendigung (Abbruch) einer Task der vorhandene gemeinsam genutzte Speicher nicht automatisch freigegeben wird.

USERDATAKEY

Gibt an, dass CICS Speicher aus einem dynamischen Benutzerschlüsselspeicherbereich (UDSA, SDSA, EUDSA oder ESDSA) zuordnet, wobei die in der Transaktionsressourcendefinition angegebene Option TASKDATAKEY überschrieben wird. Wenn Sie keine Datenschlüsseloption angeben, hängt der Speicherschlüssel (CICS-Schlüssel oder Benutzerschlüssel) von der Option TASKDATAKEY in der Transaktionsressourcendefinition ab.

Bedingungen

22 LENGERR

RESP2-Werte:

1

Der Wert von FLENGTH ist kleiner als 1 oder größer als die Länge des dynamischen Zielspeicherbereichs, aus dem die Anforderung erfüllt werden soll. Informationen zu DSAs finden Sie unter [Dynamische CICS-Speicherbereiche](#).

Diese Bedingung tritt außerdem auf, wenn der Wert von LENGTH null ist.

Standardaktion: Die Task wird abnormal beendet.

42 NOSTG

RESP2-Werte:

2

Der angeforderte Speicher ist größer als der zurzeit verfügbare Speicher im dynamischen Zielspeicherbereich. Informationen zu DSAs finden Sie unter [Dynamische CICS-Speicherbereiche](#).

Standardaktion: Die Bedingung wird ignoriert. Ein aktiver Befehl HANDLE CONDITION NOSTG löst diese Bedingung ebenfalls aus.

Beispiele

Das folgende Beispiel zeigt, wie ein 1024 Byte großer Bereich aus Benutzerschlüsselspeicher unterhalb von 16 MB (unter der Annahme, dass TASKDATAKEY(USER) in der TRANSACTION-Ressourcendefinition angegeben ist) abgerufen und mit Leerzeichen initialisiert wird:

```
EXEC CICS GETMAIN SET(PTR)
        FLENGTH(1024)
        BELOW
        INITIMG(BLANK)
```

Sie müssen BLANK in Ihrem Programm zur Darstellung eines Leerzeichens definieren.

Das folgende Beispiel zeigt, wie ein 2048 Byte großer Bereich aus CICS-Schlüsselspeicher oberhalb von 16 MB, jedoch unterhalb von 2 GB (unabhängig von der Option TASKDATAKEY, die in der Transaktionsressourcendefinition angegeben ist) abgerufen und mit Leerzeichen initialisiert wird:

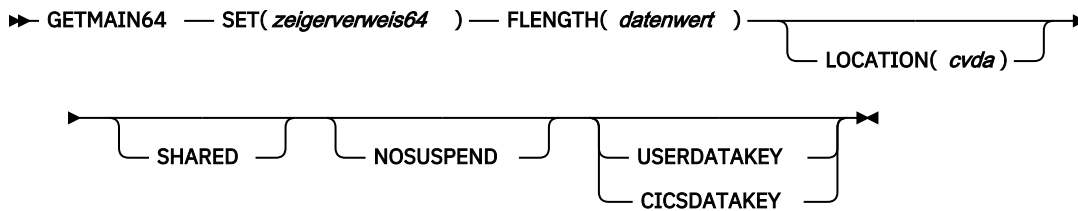
```
EXEC CICS GETMAIN SET(PTR)
        FLENGTH(2048)
        INITIMG(BLANK)
        CICS DATAKEY
```

GETMAIN64

Ruft 24-, 31- oder 64-Bit-Speicher ab. Dieser Befehl ist nur zur Verwendung in AMODE(64)-Anwendungsprogrammen in Assemblersprache vorgesehen, die nicht über Language Environment (LE) ausgeführt werden.

Weitere Informationen finden Sie unter [Beschränkungen und Anforderungen an die Assemblersprachprogrammierung](#).

GETMAIN64



Bedingungen: INVREQ, LENGERR, NOSTG

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl GETMAIN64 ruft einen Hauptspeicherbereich der Größe ab, die durch die Option FLENGTH angegeben wird. Die Adresse des Bereichs wird in dem 64-Bit-Zeigerverweis zurückgegeben, der in der Option SET angegeben wird. Standardmäßig bestimmt der Adressierungsmodus (AMODE) des aufrufenden Programms die Position des abgerufenen Speichers. Daher ruft der Befehl GETMAIN64 64-Bit-Speicher für ein Programm mit AMODE(64) ab.

Mit dem Parameter LOCATION können Sie auch unabhängig vom Adressierungsmodus des aufrufenden Programms angeben, dass 24- oder 31-Bit-Speicher abgerufen wird. Wenn Sie zum Beispiel LOCATION(LOC31) angeben, wird eine 64-Bit-Adresse für einen Bereich mit 31-Bit-Speicher zurückgegeben.

CICS ordnet Speicher immer an 16-Byte-Grenzen zu und rundet die angeforderte Länge bis zum nächsten 16-Byte-Vielfachen auf.

CICS ordnet Speicher aus einem der folgenden dynamischen Speicherbereiche (DSAs) zu:

- DSAs im 24-Bit-Speicher:
 - Der dynamische CICS-Speicherbereich (CDSA) unterhalb von 16 MB (unterhalb der 16-MB-Grenze)
 - Der dynamische Benutzerspeicherbereich (UDSA) unterhalb von 16 MB
 - Der gemeinsam genutzte dynamische Speicherbereich (SDSA) unterhalb 16 MB
- DSAs im 31-Bit-Speicher, dem erweiterten dynamischen Speicherbereich (EDSA):
 - Der erweiterte dynamische CICS-Speicherbereich (ECDSA) oberhalb von 16 MB, jedoch unterhalb von 2 GB (oberhalb der 16-MB-Grenze)
 - Der erweiterte dynamische Benutzerspeicher (EUDSA) oberhalb von 16 MB, jedoch unterhalb von 2 GB
 - Der erweiterte gemeinsam genutzte dynamische Speicherbereich (ESDSA) oberhalb von 16 MB, jedoch unterhalb von 2 GB
- DSAs im 64-Bit-Speicher: dynamischer Speicherbereich oberhalb der 2-GB-Grenze (GDSA):
 - Der dynamische CICS-Speicherbereich oberhalb der 2-GB-Grenze (GCDSA)
 - Der dynamische Benutzerspeicherbereich oberhalb der 2-GB-Grenze (GUDSA)
 - Der gemeinsam genutzte dynamische Speicherbereich oberhalb der 2-GB-Grenze (GSDSA)

Weitere Informationen zu diesen DSAs finden Sie unter [Dynamische CICS-Speicherbereiche](#).

Anmerkung: Der Befehl GETMAIN64 kann nicht zum Abrufen von Speicher aus den folgenden DSAs verwendet werden:

- Der schreibgeschützte DSA (RDSA)
- Der erweiterte schreibgeschützte DSA (ERDSA)
- Der erweiterte vertrauenswürdige DSA (ETDSA)

CICS ordnet Speicher aus einem CICS-Schlüssel oder aus dem gemeinsam genutzten DSA abhängig von den folgenden Optionen zu:

- USERDATAKEY
- CICSSTACKKEY
- SHARED
- Wenn keine Datenschlüsseloption im Befehl GETMAIN64 angegeben wird, wird die Option TASKDATAKEY in der RDO-TRANSACTION-Ressourcendefinition verwendet, unter der das anfordernde Programm ausgeführt wird. Weitere Informationen finden Sie unter [TRANSACTION-Attribute](#).

In der folgenden Tabelle sind die Auswirkungen der Datenschlüsseloptionen zusammengefasst.

Tabelle 16. Datenschlüsseloptionen im Befehl GETMAIN64		
Ohne Datenschlüsseloption	Angabe von USERDATAKEY	Angabe von CICSSTACKKEY
Speicherschlüssel wird durch TASKDATAKEY in der Transaktionsdefinition bestimmt.	Benutzerschlüsselspeicher. Wenn die Option SHARED nicht angegeben wird, wird der Speicher aus dem UD-SA, EUDSA oder GUDSA abgerufen. Wird die Option SHARED angegeben, wird der Speicher aus dem SDSA, ESDSA oder GSDSA abgerufen.	CICS-Schlüsselspeicher. Speicher wird aus dem CDSA, ECDSA oder GCDSA abgerufen.

Die Datenschlüsseloption im Befehl GETMAIN64 überschreibt die Option TASKDATAKEY in der RDO-TRANSACTION-Ressourcendefinition. Sie können zum Beispiel die Option CICSSTACKKEY angeben, um sicherzustellen, dass das anfordernde Programm CICS-Schlüsselspeicher aus einem CICS-DSA abrufen, auch wenn die Option TASKDATAKEY(USER) in der RDO-TRANSACTION-Ressourcendefinition angegeben ist.

Der Speicher, den eine Task abrufen, ist verfügbar, bis er durch einen Befehl FREEMAIN oder FREEMAIN64 freigegeben wird. Ein Bereich, der ohne die Option SHARED abgerufen wurde, kann nur durch die Task, die den Speicher abgerufen hat, freigegeben werden. Am Ende einer Task gibt CICS solchen Speicher automatisch frei, falls er noch nicht freigegeben wurde.

Jeder Speicher, der mit der Option SHARED abgerufen wird, ist für alle Tasks zugänglich, einschließlich der Tasks, die mit Transaktionsisolation ausgeführt werden. Allerdings wird ein Bereich mit der Option SHARED am Ende einer Task nicht freigegeben, sondern verbleibt, bis er explizit freigegeben wird. Jede Task kann die Anforderung FREEMAIN oder FREEMAIN64 absetzen. Dies bedeutet, dass Sie SHARED-Speicher zur Kommunikation von einer Task zur anderen verwenden können.

Anmerkung: Die Verwendung des Befehls GETMAIN64 mit der Option SHARED kann möglicherweise transaktionsübergreifende Affinitäten erstellen, die sich negativ auf die Verwendung des dynamischen Transaktionsroutings auswirken. Weitere Informationen zu Transaktionsaffinitäten finden Sie unter [Affinität](#).

Optionen

CICSSTACKKEY

Gibt an, dass CICS Speicher aus einem dynamischen CICS-Schlüsselspeicherbereich (CDSA, ECDSA oder GCDSA) zuordnet, wobei die in der Transaktionsressourcendefinition angegebene Option TASKDATAKEY überschrieben wird. Wenn Sie keine Datenschlüsseloption angeben, hängt der Speicherschlüssel (CICS-Schlüssel oder Benutzerschlüssel) von der Option TASKDATAKEY in der Transaktionsressourcendefinition ab.

Anmerkung: Wenn das Programm unter einer Task ausgeführt wird, die mit TASKDATAKEY(USER) in der Transaktionsressourcendefinition definiert ist, verwenden Sie den Befehl FREEMAIN oder FREEMAIN64 nicht explizit, sondern lassen Sie den Speicher im Rahmen der Taskbeendigung freigeben.

FLLENGTH(datenwert)

Gibt die Anzahl Byte des erforderlichen Speichers im Vollwort-Binärformat an.

Für 64-Bit-Speicher beträgt die maximale Länge, die Sie angeben können, 2146435056 (2 GB - (1 MB + 16 Byte)). Wenn die angeforderte Länge größer als der verfügbare Speicher ist, tritt die Bedingung NOSTG auf.

Für 24- oder 31-Bit-Speicher ist die maximale Länge, die Sie angeben können, der Wert der Begrenzung für den entsprechenden dynamischen Speicherbereich (DSA), das heißt DSALIMIT oder EDSALIMIT. DSALIMIT und EDSALIMIT sind Systeminitialisierungsparameter, die die Gesamtspeicherbegrenzungen definieren, innerhalb deren CICS Speicher zuordnen und die einzelnen DSAs im 24- bzw. 31-Bit-Speicher verwalten kann. Wenn die angeforderte Länge größer als der entsprechende Wert von DSALIMIT oder EDSALIMIT ist, tritt die Bedingung LENGERR auf. Wenn die angeforderte Länge kleiner als die entsprechende Begrenzung, jedoch größer als der verfügbare Speicher ist, tritt die Bedingung NOSTG auf.

LOCATION(*cvda*)

Gibt an, dass CICS Speicher aus dem 24- oder 31-Bit-Speicher unabhängig vom Adressierungsmodus (AMODE) des aufrufenden Programms zuordnen soll. Die CVDA-Werte lauten wie folgt:

LOC24

24-Bit-Speicher (unterhalb der 16 MB-Grenze) wird abgerufen, das heißt, aus dem CDSA, UDSA oder SDSA.

LOC31

31-Bit-Speicher (oberhalb der 16 MB-Grenze, aber unter 2 GB) wird abgerufen, das heißt, aus dem ECDSA, EUDSA oder ESDSA.

NOSUSPEND

Gibt an, dass CICS, wenn kein Speicher verfügbar ist, die Task nicht aussetzt, sondern die Bedingung NOSTG ausgibt.

SET(*zeigerverweis64*)

Setzt den 64-Bit-Zeigerverweis auf die Adresse des angeforderten Hauptspeichers. Der Zeiger wird auf das erste Byte des Speicherbereichs gesetzt.

Der Zeigerverweis gibt eine 64-Bit-Adresse für einen Bereich aus dem 64-, 31- oder 24-Bit-Speicher zurück. Die Position des abgerufenen Speichers hängt vom Adressierungsmodus (AMODE) des aufrufenden Programms ab, wenn nicht die Option LOCATION angegeben wird.

SHARED

Verhindert die automatische Freigabe von Speicher, der durch einen Befehl GETMAIN64 abgerufen wurde, am Ende der Task, die ihn angefordert hat. Dies ermöglicht eine Task-zu-Task-Kommunikation. Ein Bereich, der mit der Option SHARED abgerufen wurde, wird erst freigegeben, wenn ein entsprechender Befehl FREEMAN oder FREEMAIN64 entweder durch die Task, die den Speicher angefordert hat, oder durch eine andere Task abgesetzt wird.

Beachten Sie, dass bei einer abnormalen Beendigung (Abbruch) einer Task der vorhandene gemeinsam genutzte Speicher nicht automatisch freigegeben wird.

USERDATAKEY

Gibt an, dass CICS Speicher aus einem dynamischen Benutzerschlüsselspeicherbereich (UDSA, SDSA, EUDSA, ESDSA, GUDSA oder GSDSA) zuordnet, wobei die in der Transaktionsressourcendefinition angegebene Option TASKDATAKEY überschrieben wird. Wenn Sie keine Datenschlüsseloption angeben, hängt der Speicherschlüssel (CICS-Schlüssel oder Benutzerschlüssel) von der Option TASKDATAKEY in der Transaktionsressourcendefinition ab.

Bedingungen

16 INVREQ

RESP2-Werte:

3

Die Option LOCATION ist nicht gültig.

22 LENGERR

RESP2-Werte:

1

Der Wert von FLENGTH ist kleiner als 1 oder größer als die Länge des dynamischen Zielspeicherbereichs, aus dem die Anforderung erfüllt werden soll. Informationen zu DSAs finden Sie unter [Dynamische CICS-Speicherbereiche](#).

42 NOSTG

RESP2-Werte:

2

Der angeforderte Speicher ist größer als der zurzeit verfügbare Speicher im dynamischen Zielspeicherbereich. Informationen zu DSAs finden Sie unter [Dynamische CICS-Speicherbereiche](#).

Beispiel

Das folgende Beispiel zeigt, wie ein 1048576 Byte großer Bereich aus CICS-Schlüsselspeicher oberhalb der 2-GB-Grenze (unabhängig von der Option TASKDATAKEY, die in der Transaktionsressourcendefinition angegeben ist) abgerufen wird:

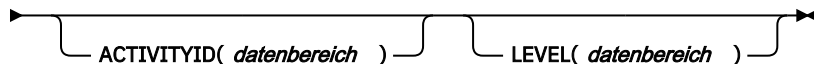
```
EXEC CICS GETMAIN64 SET(5)
      FLENGTH(1048576)
      CICSDATAKEY
```

GETNEXT ACTIVITY

Durchsucht die untergeordneten Aktivitäten einer BTS-Aktivität oder eines BTS-Prozesses.

GETNEXT ACTIVITY

➡ GETNEXT — ACTIVITY(*datenbereich*) — BROWSETOKEN(*datenwert*) →



Bedingungen: ACTIVITYERR, END, ILLOGIC, IOERR, TOKENERR

Beschreibung

Der Befehl GETNEXT ACTIVITY gibt einen der folgenden Namen mit ID zurück:

- Den Namen und die ID der nächsten untergeordneten Aktivität einer BTS-Aktivität (wenn die Optionen PROCESS und PROCESSTYPE im Befehl STARTBROWSE nicht angegeben wurden).
- Den Namen und die ID der nächsten untergeordneten Aktivität eines BTS-Prozesses (wenn die Optionen PROCESS und PROCESSTYPE im Befehl STARTBROWSE angegeben wurden).

Mithilfe des Befehls INQUIRE ACTIVITYID können Sie die angegebene Aktivität abfragen.

Optionen

ACTIVITYID(*datenbereich*)

Gibt die 52 Zeichen lange ID der nächsten Aktivität zurück.

ACTIVITY(*datenbereich*)

Gibt den 16 Zeichen langen Namen der nächsten Aktivität zurück.

BROWSETOKEN(*datenwert*)

Gibt ein Browse-Token als Vollwort-Binärwert an, das bei einem vorherigen Befehl STARTBROWSE ACTIVITY zurückgegeben wurde.

LEVEL(*datenbereich*)

Gibt einen Vollwortwert zurück, der die Tiefe des Aktivitätsbaums angibt, auf der die nächste Aktivität liegt.

Beim Durchsuchen (Browsing) der untergeordneten Aktivitäten eines Prozesses gibt der Wert '0' die Stammaktivität, der Wert '1' eine untergeordnete Aktivität der Stammaktivität, der Wert '2' eine untergeordnete Aktivität der zweiten Ebene der Stammaktivität usw. an.

Beim Durchsuchen der untergeordneten Aktivitäten einer Aktivität wird immer der Wert 0 zurückgegeben.

Bedingungen

109 ACTIVITYERR

RESP2-Werte:

19

Die Anforderung hat das Zeitlimit überschritten. Es kann sein, dass die Beendigung einer anderen Task, die diesen Aktivitätsdatensatz verwendet, verhindert wurde.

83 END

RESP2-Werte:

2

Es sind keine weiteren Ressourcendefinitionen dieses Typs vorhanden.

21 ILLOGIC

RESP2-Werte:

1

Der Wert, der in der Option BROWSETOKEN angegeben wurde, entspricht einem aktuellen Browse-Token, jedoch nicht einem, das zurzeit zum Durchsuchen einer Aktivität verwendet wird.

17 IOERR

RESP2-Werte:

29

Die Repository-Datei ist nicht verfügbar.

30

Es ist ein Ein-/Ausgabefehler für die Repository-Datei aufgetreten.

112 TOKENERR

RESP2-Werte:

3

Das Suchtoken ist ungültig.

GETNEXT CONTAINER (BTS)

Durchsucht die Container, die einer BTS-Aktivität oder einem BTS-Prozess zugeordnet sind.

GETNEXT CONTAINER

► GETNEXT — CONTAINER(*datenbereich*) — BROWSETOKEN(*datenwert*) ◄

Bedingungen: END, ILLOGIC, TOKENERR

Beschreibung

Der Befehl GETNEXT CONTAINER gibt den Namen des nächsten Containers zurück, der einer BTS-Aktivität oder einem BTS-Prozess zugeordnet ist. Mithilfe des Befehls INQUIRE CONTAINER können Sie den zurückgegebenen Container abfragen.

Anmerkung:

1. Sie können aufeinanderfolgende Befehle GETNEXT CONTAINER zum Abrufen der Namen aller Container des Prozesses oder der Aktivität verwenden, die zu dem Zeitpunkt vorhanden waren, zu dem der Befehl STARTBROWSE CONTAINER ausgeführt wurde. Jedoch werden die Namen von Containern

nicht zurückgegeben, die nach dem Befehl STARTBROWSE und vor ihrer Rückgabe durch einen Befehl GETNEXT gelöscht wurden.

2. Die Namen von Containern, die nach der Ausführung des Befehls STARTBROWSE in diesem Prozess oder dieser Aktivität erstellt werden oder in diesen Prozess bzw. diese Aktivität verschoben werden, werden möglicherweise zurückgegeben oder auch nicht zurückgegeben.
3. Die Reihenfolge, in der Container zurückgegeben werden, ist nicht definiert.

Optionen

BROWSETOKEN(datenwert)

Gibt ein Browse-Token als Vollwort-Binärwert an, das bei einem vorherigen Befehl STARTBROWSE CONTAINER zurückgegeben wurde.

CONTAINER(datenbereich)

Gibt den 16 Zeichen langen Namen des nächsten Datencontainers zurück.

Bedingungen

83 END

RESP2-Werte:

2

Es sind keine weiteren Container für diesen Prozess oder diese Aktivität vorhanden.

21 ILLOGIC

RESP2-Werte:

1

Der Wert, der in der Option BROWSETOKEN angegeben wurde, entspricht einem aktuellen Browse-Token, jedoch nicht einem, das zurzeit zum Durchsuchen von Containern verwendet wird.

112 TOKENERR

RESP2-Werte:

3

Das Suchtoken ist ungültig.

GETNEXT CONTAINER (CHANNEL)

Durchsucht (Browsing) die Container, die einem Kanal zugeordnet sind.

GETNEXT CONTAINER

➡ GETNEXT — CONTAINER(*datenbereich*) — BROWSETOKEN(*datenwert*) ➡

Bedingungen: END, ILLOGIC, TOKENERR

Beschreibung

Der Befehl **GETNEXT CONTAINER (CHANNEL)** gibt den Namen des nächsten Containers zurück, der einem Kanal zugeordnet ist.

Anmerkung:

1. Sie können aufeinanderfolgende Befehle **GETNEXT CONTAINER (CHANNEL)** zum Abrufen der Namen aller Container des Kanals verwenden, die zu dem Zeitpunkt vorhanden waren, zu dem der Befehl **STARTBROWSE CONTAINER (CHANNEL)** ausgeführt wurde. Jedoch werden die Namen von Containern nicht zurückgegeben, die nach dem Befehl **STARTBROWSE** und vor ihrer Rückgabe durch einen Befehl **GETNEXT** gelöscht wurden.
2. Die Namen von Containern, die nach der Ausführung des Befehls **STARTBROWSE** in diesem Kanal oder dieser Aktivität erstellt werden oder in diesen Kanal bzw. diese Aktivität verschoben werden, werden möglicherweise zurückgegeben oder auch nicht zurückgegeben.

3. Die Reihenfolge, in der Container zurückgegeben werden, ist nicht definiert und kann sich ändern. Anwendungen sollten sich daher nicht auf die Reihenfolge der zurückgegebenen Container stützen. Falls Sie Anwendungen nutzen, die auf diese Weise geschrieben sind, finden Sie unter [Upgrade für Anwendungen durchführen](#) wichtige Hinweise. Angaben über die bewährten Verfahren bei der Verwendung eines Kanals in Anwendungen enthält der Abschnitt [Kanal entwerfen: Bewährte Verfahren](#).

Optionen

BROWSETOKEN(datenwert)

Gibt ein Browse-Token als Vollwort-Binärwert an, das bei einem vorherigen Befehl STARTBROWSE CONTAINER (CHANNEL) zurückgegeben wurde.

CONTAINER(datenbereich)

Gibt den 16 Zeichen langen Namen des nächsten Datencontainers zurück.

Bedingungen

83 END

RESP2-Werte:

2

Es sind keine weiteren Container vorhanden.

21 ILLOGIC

RESP2-Werte:

1

Der Wert, der in der Option BROWSETOKEN angegeben wurde, entspricht einem aktuellen Browse-Token, jedoch nicht einem, das zurzeit zum Durchsuchen von Containern verwendet wird.

112 TOKENERR

RESP2-Werte:

3

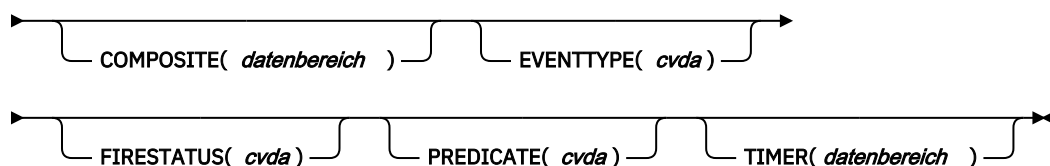
Das Suchtoken ist ungültig.

GETNEXT EVENT

Durchsucht die Ereignisse, die einer BTS-Aktivität bekannt sind.

GETNEXT EVENT

➡ GETNEXT — EVENT(*datenbereich*) — BROWSETOKEN(*datenwert*) —>



Bedingungen: END, TOKENERR

Beschreibung

Der Befehl GETNEXT EVENT gibt die Attribute des nächsten Ereignisses oder Unterereignisses zurück, das sich im Gültigkeitsbereich einer BTS-Aktivität befindet.

Optionen

BROWSETOKEN(datenwert)

Gibt ein Browse-Token als Vollwort-Binärwert an, das bei einem vorherigen Befehl STARTBROWSE EVENT zurückgegeben wurde.

COMPOSITE(datenbereich)

Wenn das angegebene Ereignis ein Unterereignis ist, gibt diese Option den 16 Zeichen langen Namen des zusammengesetzten Ereignisses zurück, das Teil des angegebenen Ereignisses ist.

EVENT(datenbereich)

Gibt den 16 Zeichen langen Namen des nächsten Ereignisses zurück. Dabei kann es sich um folgende Typen von Ereignis handeln:

- Ein atomares Ereignis. Ein atomares Ereignis, das für diesen Befehl zurückgegeben wird, kann ein Unterereignis sein oder nicht.
- Ein zusammengesetztes Ereignis.
- Ein Systemereignis.

EVENTTYPE(cvda)

Gibt den Typ des angegebenen Ereignisses an. Die CVDA-Werte lauten wie folgt:

ACTIVITY

Beendigung von Aktivitäten

COMPOSITE

Zusammengesetztes Ereignis

INPUT

Eingabeereignis

SYSTEM

Systemereignis

TIMER

Zeitgeberereignis

FIRESTATUS(cvda)

Gibt den Status des angegebenen Ereignisses an. Die CVDA-Werte lauten wie folgt:

FIRED

Das Ereignis wurde normal ausgelöst.

NOTFIRED

Das Ereignis wurde nicht ausgelöst.

PREDICATE(cvda)

Gibt den booleschen Operator an, der auf das Prädikat angewendet wird, wenn das angegebene Ereignis ein zusammengesetztes Ereignis ist. Die CVDA-Werte lauten wie folgt:

AND

Der boolesche Operator, der auf das Prädikat angewendet wird, ist AND.

OR

Der boolesche Operator, der auf das Prädikat angewendet wird, ist OR.

TIMER(datenbereich)

Wenn das angegebene Ereignis ein Zeitgeberereignis ist, gibt diese Option den 16 Zeichen langen Namen des zugeordneten Zeitgebers zurück.

Bedingungen**83 END**

RESP2-Werte:

2

Es sind keine weiteren Ressourcendefinitionen dieses Typs vorhanden.

112 TOKENERR

RESP2-Werte:

3

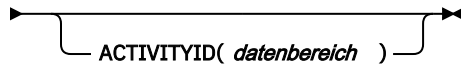
Das Suchtoken ist ungültig.

GETNEXT PROCESS

Durchsucht alle Prozesse eines angegebenen Typs in dem CICS-BTS-System (BTS - Business Transaction Services).

GETNEXT PROCESS

➡ GETNEXT — PROCESS(*datenbereich*) — BROWSETOKEN(*datenwert*) →



Bedingungen: END, ILLOGIC, IOERR, PROCESSERR, TOKENERR

Beschreibung

Der Befehl GETNEXT PROCESS gibt den Namen des nächsten Prozesses eines angegebenen Typs innerhalb des CICS-BTS-Systems zurück.

Optionen

ACTIVITYID(datenbereich)

Gibt die 52 Zeichen lange ID der Stammaktivität des nächsten Prozesses zurück.

BROWSETOKEN(datenwert)

Gibt ein Browse-Token als Vollwort-Binärwert an, das bei einem vorherigen Befehl STARTBROWSE PROCESS zurückgegeben wurde.

PROCESS(datenbereich)

Gibt den 36 Zeichen langen Namen des nächsten Prozesses zurück.

Bedingungen

83 END

RESP2-Werte:

2

Es sind keine weiteren Ressourcendefinitionen dieses Typs vorhanden.

21 ILLOGIC

RESP2-Werte:

1

Der Wert, der in der Option BROWSETOKEN angegeben wurde, entspricht einem aktuellen Browse-Token, jedoch nicht einem, das zurzeit zum Durchsuchen von Prozessen verwendet wird.

17 IOERR

RESP2-Werte:

30

Es ist ein Ein-/Ausgabefehler für die Repository-Datei aufgetreten.

108 PROCESSERR

RESP2-Werte:

13

Die Anforderung hat das Zeitlimit überschritten. Es kann sein, dass die Beendigung einer anderen Task, die diesen Prozessdatensatz verwendet, verhindert wurde.

112 TOKENERR

RESP2-Werte:

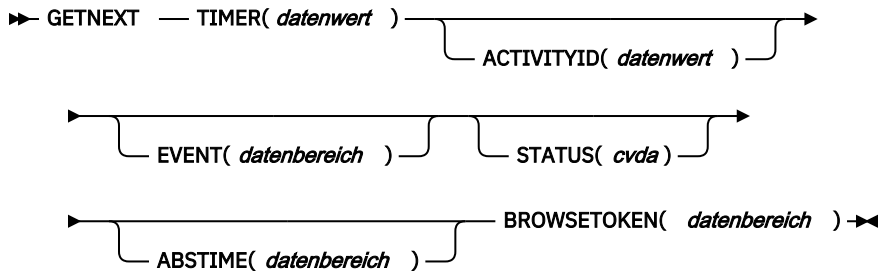
3

Das Suchtoken ist ungültig.

GETNEXT TIMER

Sucht nach den BTS-Zeitgebern, die einer BTS-Aktivität zugeordnet sind.

GETNEXT TIMER



Bedingungen: ACTIVITYERR, INVREQ, IOERR, NOTAUTH, TIMERERR

Beschreibung

Der Befehl **GETNEXT TIMER** gibt den Namen des nächsten BTS-Zeitgebers zurück, der einer BTS-Aktivität zugeordnet ist.

Optionen

ABSTIME(datenbereich)

Gibt in gepacktem Dezimalformat den Zeitpunkt für den Ablauf des Zeitgebers zurück, der in Form von Millisekunden seit 00:00 Uhr am 1. Januar 1900 ausgedrückt und auf die nächste Hundertstelsekunde gerundet wird.

Mit der Option **FORMATTIME** können Sie die Daten in andere geläufige Formate ändern.

ACTIVITYID(datenwert)

Gibt die ID (1 bis 52 Zeichen) der Aktivität an, der der Zeitgeber zugeordnet ist.

Falls diese Option nicht angegeben ist, wird von der aktuellen Aktivität ausgegangen.

BROWSETOKEN(datenbereich)

Gibt einen Datenbereich als Vollwort-Binärzahl an, in den CICS das Suchtoken schreiben soll.

EVENT(datenbereich)

Gibt den 16 Zeichen langen Namen des Ereignisses zurück, das gegebenenfalls dem Zeitgeber zugeordnet ist.

STATUS(cvda)

Gibt den Status des Zeitgebers an. Die CVDA-Werte lauten wie folgt:

EXPIRED

Der Zeitgeber ist normal abgelaufen.

FORCED

Der Ablauf des Zeitgebers wurde durch einen Befehl **FORCE TIMER** erzwungen.

UNEXPIRED

Der Zeitgeber ist noch nicht abgelaufen.

TIMER(datenwert)

Gibt den Namen (1 bis 16 Zeichen) des BTS-Zeitgebers an.

Bedingungen

109 ACTIVITYERR

RESP2-Werte:

3

Die in der Option ACTIVITYID angegebene Aktivität konnte nicht gefunden werden.

29

Die Repository-Datei ist nicht verfügbar.

30

Es ist ein Ein-/Ausgabefehler für die Repository-Datei aufgetreten.

16 INVREQ

RESP2-Werte:

1

Der Befehl wurde außerhalb des Gültigkeitsbereichs einer momentan aktiven Aktivität abgesetzt.

17 IOERR

RESP2-Werte:

30

Es ist ein Ein-/Ausgabefehler für die Repository-Datei aufgetreten.

70 NOTAUTH

RESP2-Werte:

101

Der Benutzer, der der ausgebenden Task zugeordnet ist, ist nicht berechtigt, auf die angeforderte Weise auf diese Ressource zuzugreifen.

115 TIMERERR

RESP2-Werte:

1

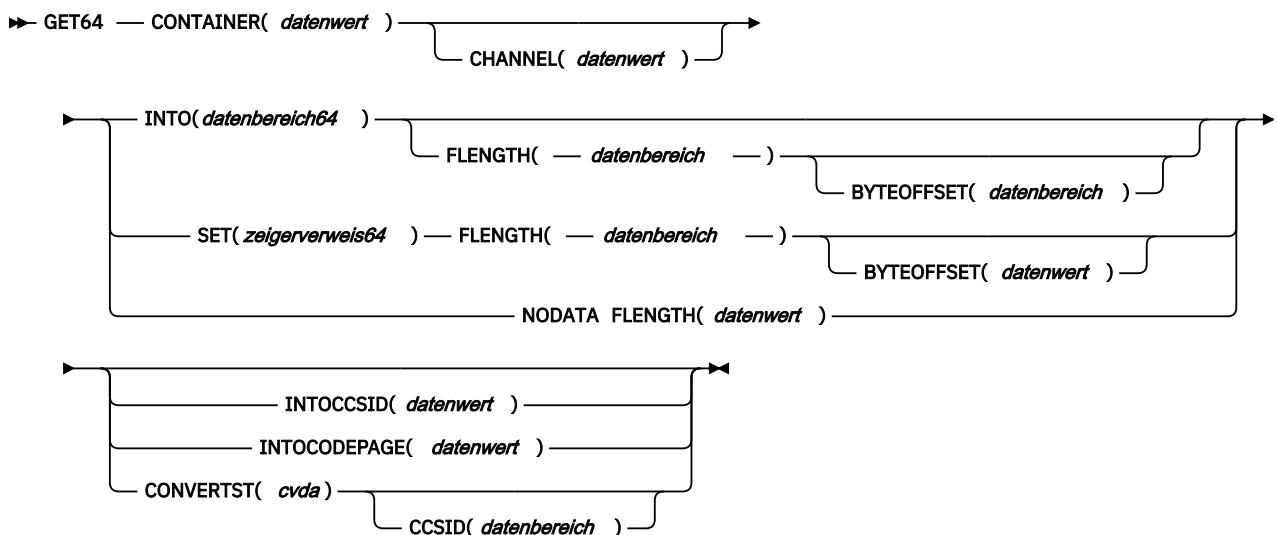
Der in der Option TIMER angegebene Zeitgeber konnte nicht gefunden werden.

GET64 CONTAINER

Ruft Daten aus einem benannten Kanalcontainer in einen 64-Bit-Speicher ab. Dieser Befehl ist nur zur Verwendung in AMODE(64)-Anwendungsprogrammen in Assemblersprache vorgesehen, die nicht über Language Environment (LE) ausgeführt werden. CICS-Geschäftstransaktionscontainer (BTS-Container) werden nicht unterstützt.

Weitere Informationen finden Sie unter [Beschränkungen und Anforderungen an die Assemblersprachprogrammierung](#).

GET64 CONTAINER



Bedingungen: CCSIDERR, CHANNELERR, CODEPAGEERR, CONTAINERERR, INVREQ, LENGERR

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl GET64 CONTAINER liest die Daten, die einem angegebenen Kanalcontainer zugeordnet sind, in 64-Bit-Speicher ein.

Der Container, der die Daten enthält, wird durch den Namen und den Kanal identifiziert, für den er ein Container ist, d. h. den Kanal, der "Eigner" des Containers ist. Der Kanal, der Eigner des Containers ist, kann explizit durch Angabe der Option CHANNEL oder implizit durch Weglassen der Option CHANNEL identifiziert werden. Wenn diese Option nicht angegeben wird, wird der aktuelle Kanal impliziert.

Optionen

BYTEOFFSET(*datenwert*)

Gibt die relative Position (Offset) in Byte an, an der die zurückgegebenen Daten beginnen. Für CHAR-Container wird der Wert von BYTEOFFSET als relative Position in den Daten in der angeforderten Codepage verwendet. Wenn Sie eine Codepage mit Mehrbytezeichen verwenden, können die zurückgegebenen Daten abhängig von dem in BYTEOFFSET angegebenen Wert Teile von Zeichen am Anfang, am Ende oder an beiden Positionen enthalten. In diesem Fall muss Ihr Anwendungsprogramm in der Lage sein, die zurückgegebenen Daten zu verarbeiten und zu interpretieren. Falls der angegebene Wert kleiner als Null ist, wird Null angenommen.

CCSID(*datenbereich*)

Gibt ein Vollwort zurück, das die ID des codierten Zeichensatzes (Coded Character Set Identifier - CCSID) der Daten enthält, die durch die Option CONVERTST(NOCONVERT) zurückgegeben werden. Sie können diese Option dazu verwenden, Container mit dem Datentyp CHAR abzurufen, ohne die Daten zu konvertieren. Wenn der Datentyp BIT für den Container angegeben wurde, ist dieser Wert null.

CHANNEL(*datenwert*)

Gibt den Namen (1 - 16 Zeichen) des Kanals an, der Eigner des Containers ist. Sie können den Kanalnamen DFHTRANSACTION angeben, um den Transaktionskanal zu verwenden.

CONTAINER(*datenwert*)

Gibt den Namen (1 - 16 Zeichen) des Containers an, der die abzurufenden Daten enthält.

CONVERTST(*cvda*)

Gibt den erforderlichen Datenkonvertierungsstatus an.

NOCONVERT

Die Containerdaten werden ohne Konvertierung abgerufen. Wenn Sie den Befehl **WEB RECEIVE** verwendet haben, um den HTTP-Hauptteil in einem Container zu speichern, und den Hauptteil unkonvertiert aus diesem Container abrufen müssen, müssen Sie die Option NOCONVERT verwenden.

FLENGTH(*datenbereich*)

Als Eingabefeld gibt FLENGTH die Länge der zu lesenden Daten als Vollwort-Binärwert an. Als Ausgabefeld gibt FLENGTH die Länge der Daten im Container zurück. FLENGTH ist abhängig von den angegebenen Optionen BYTEOFFSET, INTO, SET oder NODATA ein Eingabefeld oder ein Ausgabefeld.

Angabe der Option BYTEOFFSET

FLENGTH ist sowohl ein Eingabe- als auch ein Ausgabefeld.

Als **Eingabefeld** gibt FLENGTH die maximale Länge der Daten an, die das Programm akzeptiert. Die zurückgegebenen Daten beginnen an der relativen Position, die durch den Wert in BYTEOFFSET angegeben wird. Falls der angegebene Wert kleiner als Null ist, wird Null angenommen.

Bei Verwendung als **Ausgabefeld** (d. h. nach Abschluss der Abrufoperation) setzt CICS den Datenbereich auf die Länge der zurückgegebenen Daten. Die maximale Länge der zurückgegebenen Daten ist gleich der Länge der Daten in dem Container minus des Werts von BYTEOFFSET.

Angabe der Option INTO

FLENGTH ist sowohl ein Eingabe- als auch ein Ausgabefeld.

Als **Eingabefeld** gibt FLENGTH die maximale Länge der Daten an, die das Programm akzeptiert. Wenn der angegebene Wert kleiner null ist, wird null angenommen. Wenn die Länge der Daten den angegebenen Wert überschreitet, werden die Daten auf diesen Wert abgeschnitten und die Bedin-

gung LENGERR tritt auf. Ist die Länge der Daten kürzer als der angegebene Wert, werden die Daten kopiert, es erfolgt jedoch keine Auffüllung.

Sie müssen die Option FLENGTH nicht angeben, wenn die Länge durch den Compiler aus der Variablen INTO generiert werden kann. Wenn Sie sowohl INTO als auch FLENGTH angeben, gibt FLENGTH die maximale Länge der Daten an, die das Programm akzeptiert.

Bei Verwendung als **Ausgabefeld** (d. h. nach Abschluss der Abrufoperation) setzt CICS den Datenbereich, sofern angegeben, auf die tatsächliche Länge der Daten in dem Container. Wenn der Container Zeichendaten enthält, die von einer CCSID in eine andere konvertiert wurden, ist dies die Länge der Daten nach der Konvertierung.

Angabe der Option SET oder NODATA

FLENGTH ist nur Ausgabefeld. Es muss vorhanden sein und es muss als Datenbereich angegeben werden.

Nach Abschluss der Abrufoperation wird der Datenbereich auf die tatsächliche Länge der Daten im Container gesetzt. Wenn der Container Zeichendaten enthält, die von einer CCSID in eine andere konvertiert wurden, ist dies die Länge der Daten nach der Konvertierung.

INTO(*datenbereich64*)

Gibt den 64-Bit-Datenbereich an, in den die abgerufenen Daten eingefügt werden sollen. Der Wert *datenbereich64* bezieht sich auf einen Bereich, auf den durch einen 64-Bit-Zeiger verwiesen wird und der sich im 64-Bit-Speicher (oberhalb der 2-GB-Grenze) befinden kann.

INTOCCSID(*datenwert*)

Gibt die ID des codierten Zeichensatzes (Coded Character Set Identifier - CCSID), in den die Zeichendaten im Container konvertiert werden, als Vollwort-Binärwert an. Wenn Sie es bevorzugen, einen IANA-Namen für die Codepage anzugeben, oder wenn Sie es bevorzugen, die CCSID in Form alphanumerischer Zeichen anzugeben, verwenden Sie stattdessen die Option INTOCODEPAGE.

Für CICS Transaction Server for z/OS-Anwendungen ist die CCSID in der Regel eine EBCDIC-CCSID. Es ist jedoch möglich, eine ASCII-CCSID anzugeben, wenn Sie zum Beispiel ASCII-Daten abrufen wollen, ohne dass diese automatisch in EBCDIC konvertiert werden.

Wenn weder INTOCCSID noch INTOCODEPAGE angegeben werden, wird standardmäßig die CCSID der Region als Wert für die Konvertierung angenommen. Die Standard-CCSID der Region wird im Systeminitialisierungsparameter **LOCALCCSID** angegeben.

Es können nur Zeichendaten konvertiert werden und nur dann, wenn der Datentyp CHAR im Befehl **PUT CONTAINER** oder **PUT64 CONTAINER** angegeben wurde, der zum Einfügen der Daten in den Container verwendet wurde. Der Datentyp CHAR wird impliziert, wenn die Option FROMCCSID oder FROMCODEPAGE im Befehl **PUT CONTAINER** oder **PUT64 CONTAINER** angegeben wurde.

Weitere Informationen zur Datenkonvertierung mit Kanälen finden Sie unter [Datenkonvertierung mit Kanälen](#).

Eine Erläuterung von CCSIDs finden Sie unter [Codepagekonvertierung mit Kanälen vorbereiten](#).

INTOCODEPAGE(*datenwert*)

Gibt einen IANA-registrierten alphanumerischen Zeichensatznamen oder eine ID des codierten Zeichensatzes (CCSID) für die Codepage an, in die die Zeichendaten in dem Container konvertiert werden, wobei bis zu 40 alphanumerische Zeichen mit geeigneter Interpunktion verwendet werden. Verwenden Sie diese Option anstelle der Option CCSID, wenn Sie es bevorzugen, einen IANA-registrierten Zeichensatznamen zu verwenden, wie er im Content-Type-Header für eine HTTP-Anforderung angegeben wird. CICS konvertiert den IANA-Namen in eine CCSID. Der nachfolgende Datenkonvertierungsprozess ist identisch. Verwenden Sie diese Option außerdem dann, wenn Sie es bevorzugen, die CCSID in alphanumerischen Zeichen und nicht als Vollwort-Binärwert anzugeben.

Wenn ein IANA-Name für eine Codepage vorhanden ist und CICS die Verwendung der Codepage unterstützt, wird der Name mit der CCSID aufgelistet. Weitere Informationen finden Sie unter [Codepagekonvertierung mit Kanälen vorbereiten](#).

NODATA

Gibt an, dass keine Daten abgerufen werden. Verwenden Sie diese Option, um die Länge der Daten in dem Container (die in FLENGTH zurückgegeben wird) zu ermitteln.

Die Länge der Zeichendaten kann sich bei einer Datenkonvertierung ändern. Wenn Zeichendaten also in eine andere CCSID *als die dieser Region* konvertiert werden sollen, sollten Sie, wenn Sie NODATA angeben, auch die Option INTOCCSID angeben. Dadurch wird sichergestellt, dass die richtige Länge der konvertierten Daten in FLENGTH zurückgegeben wird.

SET(zeigerverweis64)

Gibt einen 64-Bit-Zeigerverweis an, in dem die 64-Bit-Adresse der abgerufenen Daten zurückgegeben wird. Dieser Zeigerverweis bezieht sich immer auf 64-Bit-Speicher (Speicher oberhalb der 2-GB-Grenze).

CICS verwaltet den Datenbereich, bis eines der folgenden Ereignisse auftritt:

- Ein nachfolgender Befehl **GET CONTAINER** oder **GET64 CONTAINER** mit der Option SET für denselben Container im selben Kanal wird durch ein Programm abgesetzt, das auf diesen Speicher zugreifen kann.
- Der Container wird durch einen Befehl **DELETE CONTAINER** gelöscht.
- Der Container wird durch einen Befehl **MOVE CONTAINER** verschoben.
- Der Kanal verlässt den Bereich des Programms.
- Der Kanal und die Container, die sich darin befinden, werden durch einen Befehl **DELETE CHANNEL** gelöscht.

Beachten Sie Verbindungen zu anderen Programmen, die einen dieser Befehle absetzen können.

Setzen Sie keinen Befehl **FREEMAIN64** ab, um diesen Speicher freizugeben.

Wenn Ihre Anwendung die Daten behalten muss, sollte sie sie in einen eigenen Speicher verschieben.

Bedingungen

123 CCSIDERR

RESP2-Werte:

1

Die in der Option INTOCCSID angegebene CCSID liegt außerhalb des Bereichs gültiger CCSID-Werte.

2

Die in der Option INTOCCSID angegebene CCSID und die CCSID des Containers sind eine nicht unterstützte Kombination. (Die CCSID des Containers ist der Wert, der bei der Erstellung des Containers in der Option FROMCODEPAGE oder FROMCCSID angegeben oder als Standardwert angenommen wurde.)

3

Die Daten wurden mit dem Datentyp BIT erstellt. Eine Codepagekonvertierung ist nicht möglich. Die Daten wurden ohne Codepagekonvertierung zurückgegeben.

4

Mindestens ein Zeichen konnte nicht konvertiert werden. Das Zeichen wurde durch ein Leerzeichen in den konvertierten Daten ersetzt.

5

Bei der Codepagekonvertierung eines Containers trat ein interner Fehler auf.

122 CHANNELERR

RESP2-Werte:

2

Der in der Option CHANNEL angegebene Kanal konnte nicht gefunden werden.

125 CODEPAGEERR

RESP2-Werte:

- 1 Die in der Option INTOCODEPAGE angegebene Codepage wird nicht unterstützt.
- 2 Die in der Option INTOCODEPAGE angegebene Codepage und die Codepage des Kanals sind eine nicht unterstützte Kombination.
- 3 Die Daten wurden mit dem Datentyp BIT erstellt. Eine Codepagekonvertierung ist nicht möglich. Die Daten wurden ohne Codepagekonvertierung zurückgegeben.
- 4 Mindestens ein Zeichen konnte nicht konvertiert werden. Das Zeichen wurde durch ein Leerzeichen in den konvertierten Daten ersetzt.
- 5 Bei der Codepagekonvertierung eines Containers trat ein interner Fehler auf.

110 CONTAINERERR

RESP2-Werte:

- 10 Der in der Option CONTAINER angegebene Container konnte nicht gefunden werden.

16 INVREQ

RESP2-Werte:

- 2 Die Option INTOCCSID wurde ohne die Option CHANNEL angegeben und es ist kein aktueller Kanal vorhanden (weil an das Programm, das den Befehl abgesetzt hat, kein Kanal übergeben wurde). Die Option INTOCCSID ist nur für Befehle GET64 CONTAINER gültig, die einen Kanal (explizit oder implizit) angeben.
- 4 Die Option CHANNEL wurde nicht angegeben und es ist kein aktueller Kanal vorhanden (weil an das Programm, das den Befehl abgesetzt hat, kein Kanal übergeben wurde).
- 5 Der CVDA-Wert für CONVERTST ist ungültig.

22 LENGERR

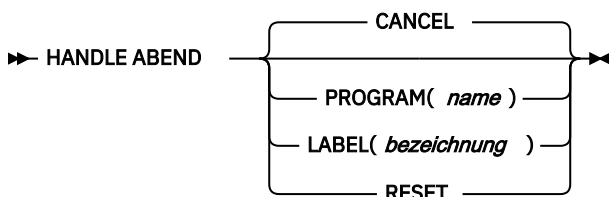
RESP2-Werte:

- 11 Die Länge des Programmbereichs ist kürzer als die Länge der Daten in dem Container. Wenn der Bereich kleiner ist, werden die Daten abgeschnitten, um sie in den Bereich einzupassen.
- 12 Die relative Position (Offset) ist größer oder gleich der Länge des Containers.

HANDLE ABEND

Verarbeitet einen Exit bei abnormaler Beendigung.

HANDLE ABEND



Bedingungen: NOTAUTH, PGMIDERR (nur PROGRAM)

Dieser Befehl ist threadsicher.

Beschreibung

Verwenden Sie den Befehl **HANDLE ABEND**, um einen Exit zur Verarbeitung einer abnormalen Beendigung zu aktivieren, abzubrechen oder zu reaktivieren. Sie können den Befehl mit den Befehlen **PUSH HANDLE** und **POP HANDLE** aussetzen. Weitere Informationen finden Sie unter [Befehle PUSH HANDLE und POP HANDLE verwenden](#).

Wenn eine Task abnormal beendet wird, sucht CICS nach einem aktiven Abbruchexit, und zwar beginnend auf der logischen Ebene des Anwendungsprogramms, in dem die abnormale Beendigung (Abbruch) aufgetreten ist, und mit Fortsetzung auf aufeinanderfolgend höheren Ebenen. Die Steuerung wird an den ersten gefundenen aktiven Abbruchexit übergeben, sofern vorhanden.

Der Befehl **HANDLE ABEND** kann keine abnormalen Beendigungen abfangen, die mit der Option CANCEL abgesetzt werden. Einige interne abnormale Beendigungen, die von CICS generiert werden, werden mit der Option CANCEL abgesetzt, wie zum Beispiel solche mit den Abbruchcodes ASPx oder APSJ.

Wenn die Bezeichnung, die in einem Befehl **HANDLE ABEND LABEL** angegeben wird, die Steuerung empfängt, werden die Register wie folgt gesetzt:

COBOL

Die Steuerung wird an den Befehl **HANDLE ABEND** mit den wiederhergestellten Registern zurückgegeben. Anschließend wird die Anweisung COBOL GO TO ausgeführt.

Assembler

R15: Abbruchbezeichnung. R0-14: Inhalt an dem Punkt, zu dem der letzte EXEC CICS-Befehl auf derselben Programmebene wie der Befehl **HANDLE ABEND** abgesetzt wurde.

Wenn LABEL angegeben wird, werden der Adressierungsmodus und der Ausführungsschlüssel des Programms verwendet, das den Befehl **HANDLE ABEND** abgesetzt hat.

Wenn PROGRAM angegeben wird, wird der Adressierungsmodus durch die Art und Weise definiert, in der die Links des Programms bearbeitet wurden, und der Ausführungsschlüssel wird durch die Option EXECKEY in der Ressourcendefinition des Programms angegeben.

Wenn ein Kommunikationsbereich (COMMAREA) eingerichtet wurde, wird er an das angegebene Programm (PROGRAM) übergeben. Wenn mehr als ein Anwendungsprogramm an der Task beteiligt war, ist der Kommunikationsbereich, der an den Abbruchexit übergeben wird, der Kommunikationsbereich des Programms, das den Befehl **HANDLE ABEND** abgesetzt hat. Dies ist möglicherweise nicht der Kommunikationsbereich des Programms, in dem der Abbruch aufgetreten ist.

Wenn ein aktueller Kanal vorhanden ist, ist er über das angegebene Programm (PROGRAM) zugänglich.

Optionen

CANCEL

Gibt an, dass ein zuvor eingerichteter Exit auf der logischen Ebene des Anwendungsprogramms, das die Steuerung innehat, inaktiviert wird. Dies ist die Standardoption.

LABEL(*bezeichnung*)

Gibt die Programmbezeichnung an, an die die Steuerung verzweigt, wenn eine abnormale Beendigung auftritt.

Sie können diese Option nicht für Anwendungsprogramme in Assemblersprache mit AMODE(64), C, C++ oder PL/I verwenden.

PROGRAM(*name*)

Gibt den Namen des Programms an, an das die Steuerung übergeben wird, wenn die Task abnormal beendet wird. Wenn die Abbruchbedingung ausgelöst wird und das angegebene Programm nicht bereits definiert ist, wird dieses Programm automatisch installiert.

Das in dieser Option angegebene Programm sollte immer mit einem Abbruch (ABEND) beendet werden, außer wenn es Abbrüche behandelt, die infolge von Anwendungsprogrammlogik generiert werden.

RESET

Gibt an, dass ein Exit, der durch einen Befehl HANDLE ABEND CANCEL oder durch CICS inaktiviert wurde, reaktiviert wird.

Diese Option wird in der Regel durch eine Routine eines Exits für abnormale Beendigung abgesetzt.

Bedingungen

70 NOTAUTH

Tritt auf, wenn eine Ressourcensicherheitsprüfung für PROGRAM(name) fehlgeschlagen ist.

Standardaktion: Die Task wird abnormal beendet.

27 PGMIDERR

RESP2-Werte:

1

Das Programm hat keine installierte Ressourcendefinition und die automatische Installation für Programme ist nicht aktiv.

2

Das Programm ist inaktiviert.

9

Die installierte Programmressourcendefinition bezieht sich auf ein fernes Programm.

Standardaktion: Die Task wird abnormal beendet.

Beispiele

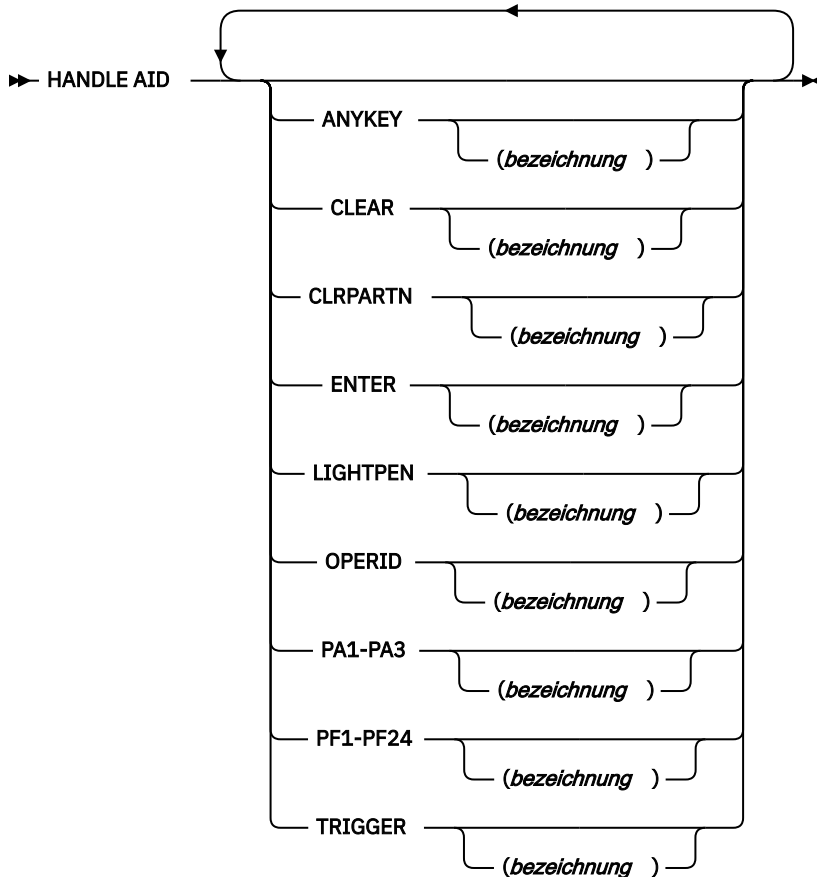
Das folgende Beispiel zeigt, wie ein Programm als Exit eingerichtet wird:

```
EXEC CICS HANDLE ABEND  
      PROGRAM('EXITPGM')
```

HANDLE AID

Verarbeitet AID-Zeichen (AID - Attention Identifiers).

HANDLE AID



Bedingung: INVREQ

Dieser Befehl ist threadsicher.

Beschreibung

Mit dem Befehl HANDLE AID können Sie die Bezeichnung angeben, an die die Steuerung übergeben werden soll, wenn ein AID-Zeichen von einer Bildschirmereinheit empfangen wird. Die Steuerung wird übergeben, nachdem der Eingabebefehl abgeschlossen wurde, das heißt, nachdem alle empfangenen Daten zusätzlich zum AID-Zeichen an das Anwendungsprogramm übergeben wurden.

Einschränkung: Dieser Befehl wird nur in Anwendungen unterstützt, die in COBOL, PL/I und Assemblersprache geschrieben sind (jedoch nicht in Anwendungen in Assemblersprache mit AMODE(64)). In allen anderen unterstützten höheren Programmiersprachen wird der Befehl nicht unterstützt.

Wenn ein AID-Zeichen ignoriert werden soll, setzen Sie einen Befehl HANDLE AID ab, der die zugeordnete Option **ohne** Bezeichnung angibt. Dadurch wird die Wirkung dieser Option in einem beliebigen zuvor abgesetzten Befehl HANDLE AID inaktiviert.

Wenn keine Befehle HANDLE AID aktiv sind, das heißt, es wurden keine solchen Befehle abgesetzt oder alle inaktiviert, wird die Steuerung bei der Anweisung, die unmittelbar auf den Eingabebefehl folgt, an das Anwendungsprogramm zurückgegeben. Prüfen Sie das Feld EIBAID, um zu ermitteln, welche Taste gedrückt wurde.

Sie können die folgenden Optionen angeben:

- ANYKEY (beliebige Programmabruf Taste (PA), Funktionstaste (PF) oder Löschtaste (CLEAR), jedoch nicht die Eingabetaste (ENTER))
- CLEAR (für die Taste dieses Namens (Löschen))
- CLRPARTN (für die Taste dieses Namens (Partition löschen))

- ENTER (für die Taste dieses Namens (Eingabe))
- LIGHTPEN (für Lichtstiftabruf)
- OPERID (für Operator-ID-Kartenleser, den Magnetcodeleser (MSR) oder den erweiterten Magnetcodeleser (MSRE))
- PA1, PA2 oder PA3 (beliebige der Programmabruf Tasten)
- PF1 bis PF24 (beliebige der Funktionstasten)
- TRIGGER (für Triggerfeldabruf)

Sie können nicht mehr als 16 Optionen in ein und denselben Befehl einschließen.

Wenn eine Task über ein Terminal durch ein AID-Zeichen initialisiert wird, liest der erste Befehl RECEIVE in der Task nicht vom Terminal, sondern kopiert nur den Eingabepuffer (auch wenn die Länge der Daten null ist), sodass die Steuerung durch einen Befehl HANDLE AID für dieses AID-Zeichen übergeben werden kann.

Informationen zur AID-Standardliste (DFHAID) und zur Standardliste für Attribute und Druckersteuerzeichen (DFHBMSCA) finden Sie in [Kapitel 8, „BMS-spezifische Konstanten“](#), auf Seite 841.

Die Bezeichnung empfängt die Steuerung in demselben Ausführungsschlüssel, in dem das Programm ausgeführt wurde, als der Befehl HANDLE AID abgesetzt wurde.

Eine Drucktaste, die durch den Systeminitialisierungsparameter PRINT angegeben wird, hat Vorrang vor einem Befehl HANDLE AID.

Bedingungen

16 INVREQ

RESP2-Werte:

200

Der Befehl wurde von einer Distributed Program Link-Serveranwendung abgesetzt.

Standardaktion: Die Task wird abnormal beendet.

Beispiel

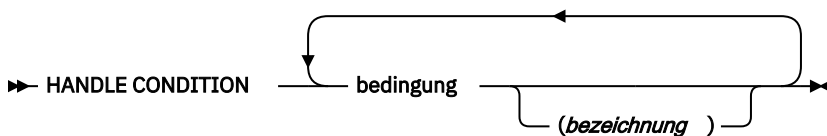
Das folgende Beispiel zeigt einen Befehl HANDLE AID, der eine Bezeichnung (Label) für die Taste PA1 und eine zweite Bezeichnung für die Tasten CLEAR, PA2, PA3 und alle Funktionstasten mit Ausnahme von PF10 angibt. Wenn ein AID-Zeichen für die Taste PF10 empfangen wird oder die Eingabetaste (ENTER) gedrückt wird, wird die Steuerung an die Anweisung unmittelbar nach dem Eingabebefehl in dem Anwendungsprogramm zurückgegeben.

```
EXEC CICS HANDLE AID PA1(LAB1)
      ANYKEY(LAB2) PF10
```

HANDLE CONDITION

Behandelt Bedingungen.

HANDLE CONDITION



Dieser Befehl ist threadsicher.

Beschreibung

Mit dem Befehl **HANDLE CONDITION** geben Sie die Bezeichnung an, an die die Steuerung übergeben werden soll, falls eine Bedingung auftritt. Sie müssen den Namen der Bedingung und optional eine Bezeichnung einschließen, an die die Steuerung übergeben werden soll, falls die Bedingung auftritt.

Einschränkung: Dieser Befehl wird nur in Anwendungen unterstützt, die in COBOL, PL/I und Assemblersprache geschrieben sind (jedoch nicht in Anwendungen in Assemblersprache mit AMODE(64)). In allen anderen unterstützten höheren Programmiersprachen wird der Befehl nicht unterstützt.

Wenn Sie den Parameter für die Bezeichnung nicht angeben, wird jeder Befehl **HANDLE CONDITION** für die Bedingung inaktiviert und es wird die Standardaktion ausgeführt, falls die Bedingung auftritt. Dies erfolgt unabhängig von der Einstellung der allgemeinen Einstellung für ERROR.

Sie müssen sicherstellen, dass der Befehl **HANDLE CONDITION** vor dem Befehl ausgeführt wird, der zu der angegebenen Bedingung führen kann.

Sie können nicht mehr als 16 Bedingungen in ein und denselben Befehl einschließen. Die Bedingungen müssen durch mindestens ein Leerzeichen voneinander getrennt angegeben werden. Sie müssen alle zusätzlichen Bedingungen in weiteren Befehlen **HANDLE CONDITION** angeben.

Wenn eine Bedingung auftritt, die nicht in einem Befehl **HANDLE CONDITION** oder **IGNORE CONDITION** angegeben ist, wird die Standardaktion ausgeführt. Wenn die Standardaktion für eine solche Bedingung die Task jedoch abnormal beendet und die Bedingung ERROR angegeben wurde, wird die Aktion für ERROR ausgeführt.

Die Bezeichnung empfängt die Steuerung in demselben Ausführungsschlüssel, in dem das Programm ausgeführt wurde, als der Befehl **HANDLE CONDITION** abgesetzt wurde.

Wenn CICS eine Bedingung behandelt, wird die Programmmaske der Anwendung in dem Wert wiederhergestellt, den sie hatte, als der Befehl EXEC CICS HANDLE CONDITION abgesetzt wurde.

Gültigkeitsbereich

Der Befehl **HANDLE CONDITION** für eine bestimmte Bedingung gilt nur für das Programm, in dem er angegeben wird. Der Befehl **HANDLE CONDITION** bleibt aktiv, während das Programm ausgeführt wird oder bis eine der folgenden Situationen auftritt:

- Ein Befehl **IGNORE CONDITION** für dieselbe Bedingung wird angetroffen. Der Befehl **HANDLE CONDITION** wird überschrieben.
- Ein weiterer Befehl **HANDLE CONDITION** für dieselbe Bedingung wird angetroffen. Der neue Befehl überschreibt den vorherigen Befehl.
- Der Befehl **LINK** wird ausgeführt, um ein anderes CICS-Programm aufzurufen. Die Optionen von **HANDLE CONDITION** werden von dem verknüpften Programm nicht übernommen.

Der Befehl **HANDLE CONDITION** wird durch die Option NOHANDLE oder RESP in einem Befehl vorübergehend inaktiviert.

Hinweise zu Programmiersprachen

Wenn in einem Anwendungsprogramm in Assemblersprache durch eine Bedingung eine Verzweigung zu einer Bezeichnung bewirkt wird, werden die Register im Anwendungsprogramm mit ihren Werten im Programm wiederhergestellt, die sie zu dem Zeitpunkt hatten, als der Befehl, der die Bedingung verursacht hat, abgesetzt wurde.

In einem PL/I-Anwendungsprogramm führt eine Verzweigung zu einer Bezeichnung in einer inaktiven Prozedur oder in einem inaktiven Begin-Block, die durch eine Bedingung verursacht wird, zu unvorhersehbaren Ergebnissen.

Optionen

bedingung(*bezeichnung*)

Gibt den Namen der Bedingung an. Der Wert *bezeichnung* gibt die Position im Programm an, zu der verzweigt werden soll, wenn die Bedingung auftritt.

Weitere Informationen zu den Bedingungen finden Sie unter [EIB-Felder](#).

Beispiele

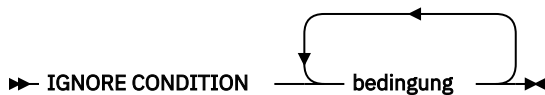
Das folgende Beispiel zeigt, wie Bedingungen wie DUPREC und LENGERR behandelt werden, die auftreten können, wenn Sie den Befehl WRITE zum Hinzufügen eines Datensatzes in einer Datei (Dataset) verwenden. Die Bedingung DUPREC wird als Sonderfall behandelt. Für die Bedingung LENGERR wird die Standardaktion (d. h. die Task wird abnormal beendet) ausgeführt. Alle anderen Bedingungen werden von der Fehlerroutine ERRHANDL behandelt.

```
EXEC CICS HANDLE CONDITION  
      ERROR(ERRHANDL)  
      DUPREC(DUPRTN) LENGERR
```

IGNORE CONDITION

Ignoriert Bedingungen.

IGNORE CONDITION



Dieser Befehl ist threadsicher.

Beschreibung

Mit dem Befehl **IGNORE CONDITION** können Sie angeben, dass keine Aktion ausgeführt wird, falls eine Bedingung auftritt (d. h., die Steuerung wird an die Anweisung zurückgegeben, die auf den Befehl folgt, dessen Ausführung fehlgeschlagen ist, und der EIB wird festgelegt). Die Ausführung eines Befehls kann dazu führen, dass verschiedene Bedingungen ausgelöst werden. CICS prüft die Bedingungen in einer vorbestimmten Reihenfolge. Nur die erste Bedingung, die (von Ihrem Befehl **IGNORE CONDITION**) nicht ignoriert wird, wird an Ihr Anwendungsprogramm übergeben.

Einschränkung: Dieser Befehl wird nur in Anwendungen unterstützt, die in COBOL, PL/I und Assemblersprache geschrieben sind (jedoch nicht in Anwendungen in Assemblersprache mit AMODE(64)). In allen anderen unterstützten höheren Programmiersprachen wird der Befehl nicht unterstützt.

Informationen zu den Bedingungen finden Sie unter [EIB-Felder](#).

Der Befehl **IGNORE CONDITION** für eine angegebene Bedingung gilt nur für das Programm, in dem er angegeben wird, und er bleibt so lange aktiv, wie das Programm ausgeführt wird oder bis ein Befehl **HANDLE CONDITION** für dieselbe Bedingung angetroffen wird, durch den der Befehl **IGNORE CONDITION** überschrieben wird.

Sie können nicht mehr als 16 Bedingungen in ein und denselben Befehl einschließen. Die Bedingungen müssen durch mindestens ein Leerzeichen voneinander getrennt angegeben werden. Sie können zusätzliche Bedingungen in weiteren Befehlen **IGNORE CONDITION** angeben.

Optionen

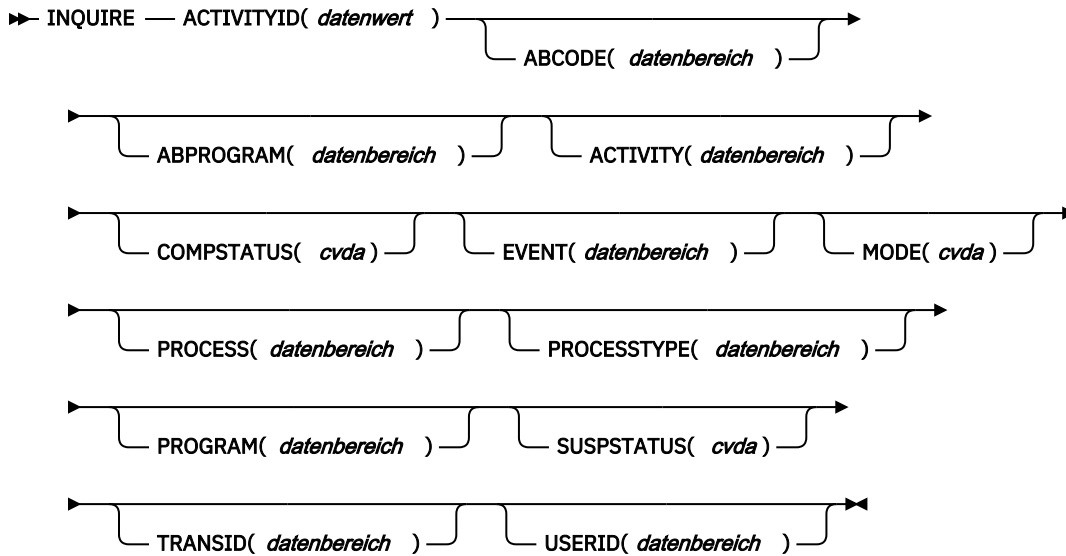
bedingung

Gibt den Namen der Bedingung an, die ignoriert werden soll.

INQUIRE ACTIVITYID

Ruft die Attribute einer BTS-Aktivität ab.

INQUIRE ACTIVITYID



Bedingungen: ACTIVITYERR, NOTAUTH

Beschreibung

Der Befehl **INQUIRE ACTIVITYID** gibt die Attribute einer angegebenen BTS-Aktivität zurück.

Mit diesem Befehl können Sie Details einer Aktivität abrufen, deren ID bei einer Browseoperation abgerufen wurde.

Optionen

ABCODE(datenbereich)

Gibt den vier Zeichen langen Abbruchcode zurück, wenn die Aktivität abnormal beendet wurde.

ABPROGRAM(datenbereich)

Gibt den acht Zeichen langen Namen des Programms zurück, das die Steuerung zum Zeitpunkt der abnormalen Beendigung innehatte, wenn die Aktivität abnormal beendet wurde.

ACTIVITY(datenbereich)

Gibt den 16 Zeichen langen Namen der Aktivität zurück, die abgefragt wird.

ACTIVITYID(datenwert)

Gibt die ID (1 - 52 Zeichen) der Aktivität an, die abgefragt werden soll. (Die Aktivitäts-ID wird in der Regel durch einen Befehl **GETNEXT ACTIVITY** während des Durchsuchens einer Aktivität abgerufen worden sein.)

COMPSTATUS(cvda)

Gibt den Beendigungsstatus der Aktivität an. Die CVDA-Werte lauten wie folgt:

ABEND

Das Programm, das die Aktivität implementiert, wurde abnormal beendet. Alle untergeordneten Aktivitäten der Aktivität wurden abgebrochen.

FORCED

Die Aktivität wurde zur Beendigung gezwungen, zum Beispiel indem sie durch einen Befehl **CANCEL ACTIVITY** abgebrochen wurde.

INCOMPLETE

Die angegebene Aktivität ist unvollständig. Dies kann Folgendes bedeuten:

- Sie wurde noch nicht ausgeführt.
- Sie ist von einer oder mehreren Aktivierungen zurückgekehrt, muss jedoch erneut aktiviert (angeschlossen) werden, um alle ihre Verarbeitungsschritte abzuschließen.
- Sie ist zurzeit aktiv.

NORMAL

Die angegebene Aktivität wurde erfolgreich abgeschlossen.

EVENT(datenbereich)

Gibt den 16 Zeichen langen Namen des Beendigungsereignisses zurück, das an den Anforderer (Requester) dieser Aktivität gesendet wird, wenn die Aktivität asynchron zum Anforderer abgeschlossen wird.

MODE(cvda)

Gibt den aktuellen Status (Modus) der Aktivität an. Die CVDA-Werte lauten wie folgt:

ACTIVE

Eine Aktivierung der Aktivität wird gerade ausgeführt.

CANCELLING

CICS wartet darauf, die Aktivität abubrechen. Es wurde ein Befehl CANCEL ACTIVITY abgesetzt, jedoch kann CICS die Aktivität nicht sofort abbrechen, weil eine oder mehrere der untergeordneten Aktivitäten der Aktivität nicht zugänglich sind.

Es werden keine weiteren Operationen an der Aktivität zugelassen, bis sie abgebrochen wurde.

COMPLETE

Die Aktivität wurde erfolgreich oder nicht erfolgreich abgeschlossen. Der Wert, der in der Option COMPSTATUS zurückgegeben wird, gibt an, wie sie abgeschlossen wurde.

DORMANT

Die Aktivität wartet darauf, dass ein Ereignis ihre nächste Aktivierung auslöst.

INITIAL

Es wurde noch kein Befehl RUN oder LINK für die Aktivität abgesetzt oder die Aktivität wurde durch einen Befehl RESET ACTIVITY zurückgesetzt.

PROCESS(datenbereich)

Gibt den 36 Zeichen langen Namen des Prozesses zurück, zu dem diese Aktivität gehört.

PROCESSTYPE(datenbereich)

Gibt den acht Zeichen langen Namen des Prozesstyps zurück, zu dem der Prozess gehört, der diese Aktivität enthält.

PROGRAM(datenbereich)

Gibt den acht Zeichen langen Namen des Programms zurück, das ausgeführt wird, wenn diese Aktivität ausgeführt wird.

SUSPSTATUS(cvda)

Gibt an, ob die Aktivität zurzeit ausgesetzt ist. Die CVDA-Werte lauten wie folgt:

SUSPENDED

Die Aktivität ist ausgesetzt. Wenn ein Reaktivierungsereignis (erneuter Anschluss) auftritt, wird sie nicht reaktiviert.

NOTSUSPENDED

Die Aktivität ist zurzeit nicht ausgesetzt. Wenn ein Reaktivierungsereignis auftritt, wird sie reaktiviert.

TRANSID(datenbereich)

Gibt die vier Zeichen lange Transaktions-ID zurück, unter der diese Aktivität ausgeführt wird.

USERID(datenbereich)

Gibt die acht Zeichen lange ID des Benutzers zurück, unter dessen Berechtigung diese Aktivität ausgeführt wird.

Bedingungen

109 ACTIVITYERR

RESP2-Werte:

1

Die in der Option ACTIVITYID angegebene Aktivitäts-ID bezieht sich auf keine Aktivität, die sich im Gültigkeitsbereich dieser Task befindet.

19

Die Anforderung hat das Zeitlimit überschritten. Es kann sein, dass die Beendigung einer anderen Task, die diesen Aktivitätsdatensatz verwendet, verhindert wurde.

29

Die Repository-Datei ist nicht verfügbar.

30

Es ist ein Ein-/Ausgabefehler für die Repository-Datei aufgetreten.

70 NOTAUTH

RESP2-Werte:

101

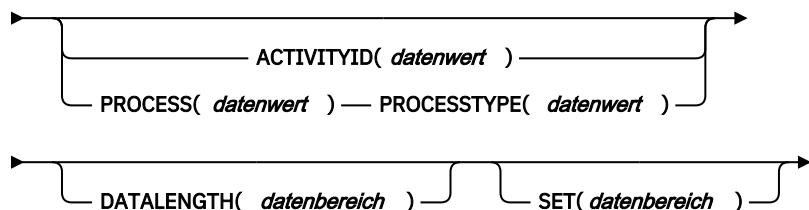
Der Benutzer, der der absetzenden Task zugeordnet ist, ist nicht berechtigt, auf diese Ressource in der angeforderten Weise zuzugreifen.

INQUIRE CONTAINER

Ruft die Attribute eines BTS-Datencontainers ab.

INQUIRE CONTAINER

➔ INQUIRE — CONTAINER(*datenwert*) ➔



Bedingungen: ACTIVITYERR, CONTAINERERR, IOERR, NOTAUTH, PROCESSERR

Beschreibung

Der Befehl INQUIRE CONTAINER gibt einen Zeiger auf den Inhalt eines angegebenen BTS-Datencontainers sowie die Länge der Daten zurück.

Zum Abfragen eines Containers, der der aktuellen Aktivität zugeordnet ist, lassen Sie die Optionen ACTIVITYID und PROCESS weg.

Zum Abfragen eines Containers, der einer anderen Aktivität zugeordnet ist, geben Sie die Option ACTIVITYID an. (Die Aktivitäts-ID, die in der Option ACTIVITYID angegeben wird, kann zum Beispiel durch einen Befehl GETNEXT ACTIVITY während einer Browseoperation zurückgegeben worden sein.)

Zum Abfragen eines Prozesscontainers (einschließlich des Containers, der dem *aktuellen* Prozess zugeordnet ist) geben Sie die Optionen PROCESS und PROCESSTYPE an.

Anmerkung:

1. Das Abfragen eines Containers der aktuellen Aktivität gibt Details der im Speicher befindlichen Version und nicht der im Repository festgeschriebenen Version zurück. Dies bedeutet, dass folgende Containerdetails sichtbar werden können:

- Container, die sich noch nicht im Repository befinden.
 - Containerinhalte, die sich von denen im Repository unterscheiden.
2. Das Abfragen eines Containers, dessen Eigner nicht die aktuelle Aktivität ist, gibt Details der im Repository festgeschriebenen Version zurück. Allerdings ist das Lesen des Repository-Datensatzes vorläufig ("dirty", da der Datensatz nicht gesperrt wird. Wenn der Datensatz gerade durch eine andere Task aktualisiert wird, sind die zurückgegebenen Daten möglicherweise nicht zuverlässig.

Optionen

ACTIVITYID(datenwert)

Gibt die ID (1-52 Zeichen) der Aktivität an, der der Datencontainer zugeordnet ist.

Wenn weder diese Option noch die Prozessoptionen angegeben werden, wird die aktuelle Aktivität angenommen.

CONTAINER(datenwert)

Gibt den Namen (1-16 Zeichen) des Datencontainers an, der abgefragt werden soll.

DATALENGTH(datenbereich)

Gibt den Vollwortwert für die Länge der Daten zurück, die in dem angegebenen Datencontainer enthalten sind.

PROCESS(datenwert)

Gibt den Namen (1-36 Zeichen) des Prozesses an, dem der Datencontainer zugeordnet ist.

Wenn weder diese Option noch die Option ACTIVITYID angegeben werden, wird die aktuelle Aktivität angenommen.

PROCESSTYPE(datenwert)

Gibt den Prozesstyp (1-8 Zeichen) des Prozesses an, der in der Option PROCESS angegeben wird.

SET(datenbereich)

Gibt einen Zeiger auf den Inhalt des Datencontainers zurück.

Bedingungen

109 ACTIVITYERR

RESP2-Werte:

2

Die in der Option ACTIVITYID angegebene Aktivität konnte nicht gefunden werden.

3

Da weder die Option ACTIVITYID noch die Optionen für den Prozess angegeben wurden, wurde eine Abfrage für die aktuelle Aktivität impliziert, jedoch ist der Anforderung keine aktuelle Aktivität zugeordnet.

29

Die Repository-Datei ist nicht verfügbar.

30

Es ist ein Ein-/Ausgabefehler für die Repository-Datei aufgetreten.

110 CONTAINERERR

RESP2-Werte:

1

Der in der Option CONTAINER angegebene Container konnte nicht gefunden werden.

17 IOERR

RESP2-Werte:

30

Es ist ein Ein-/Ausgabefehler für die Repository-Datei aufgetreten.

70 NOTAUTH

RESP2-Werte:

101

Der Benutzer, der der absetzenden Task zugeordnet ist, ist nicht berechtigt, auf diese Ressource in der angeforderten Weise zuzugreifen.

108 PROCESSERR

RESP2-Werte:

2

Der in der Option PROCESSTYPE angegebene Prozesstyp konnte nicht gefunden werden.

4

Der in der Option PROCESS angegebene Prozess konnte nicht gefunden werden.

13

Die Anforderung hat das Zeitlimit überschritten. Es kann sein, dass die Beendigung einer anderen Task, die diesen Prozessdatensatz verwendet, verhindert wurde.

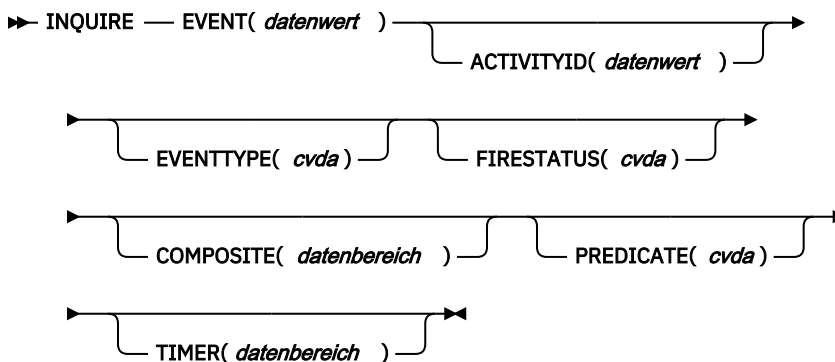
33

Der in der Option PROCESS angegebene Prozess wurde nicht festgeschrieben.

INQUIRE EVENT

Ruft die Attribute eines BTS-Ereignisses ab.

INQUIRE EVENT



Bedingungen: ACTIVITYERR, EVENTERR, INVREQ, IOERR, NOTAUTH

Beschreibung

Der Befehl `INQUIRE EVENT` gibt die Attribute eines angegebenen BTS-Ereignisses zurück.

Zum Abfragen eines Ereignisses, das der aktuellen Aktivität zugeordnet ist, lassen Sie die Option `ACTIVITYID` weg. Zum Abfragen eines Ereignisses, das einer anderen Aktivität zugeordnet ist, geben Sie die Option `ACTIVITYID` an. (Die Aktivitäts-ID, die in der Option `ACTIVITYID` angegeben wird, kann zum Beispiel durch einen Befehl `GETNEXT ACTIVITY` während einer Browseoperation zurückgegeben worden sein.)

Optionen

ACTIVITYID(datenwert)

Gibt die ID (1 - 52 Zeichen) der Aktivität an, der das Ereignis zugeordnet ist.

Wenn diese Option nicht angegeben wird, wird die aktuelle Aktivität angenommen.

COMPOSITE(datenbereich)

Wenn das angegebene Ereignis ein Unterereignis ist, gibt diese Option den 16 Zeichen langen Namen des zusammengesetzten Ereignisses zurück, das Teil des angegebenen Ereignisses ist.

EVENT(datenwert)

Gibt den Namen (1 - 16 Zeichen) des Ereignisses an, das abgefragt werden soll.

EVENTTYPE(cvda)

Gibt den Typ des angegebenen Ereignisses an. Die CVDA-Werte lauten wie folgt:

ACTIVITY

Beendigung von Aktivitäten

COMPOSITE

Zusammengesetztes Ereignis

INPUT

Eingabeereignis

SYSTEM

Systemereignis

TIMER

Zeitgeberereignis

FIRESTATUS(cvda)

Gibt den Status des angegebenen Ereignisses an. Die CVDA-Werte lauten wie folgt:

FIRED

Das Ereignis wurde normal ausgelöst.

NOTFIRED

Das Ereignis wurde nicht ausgelöst.

PREDICATE(cvda)

Gibt den booleschen Operator an, der auf das Prädikat angewendet wird, wenn das angegebene Ereignis ein zusammengesetztes Ereignis ist. Die CVDA-Werte lauten wie folgt:

AND

Der boolesche Operator, der auf das Prädikat angewendet wird, ist AND.

OR

Der boolesche Operator, der auf das Prädikat angewendet wird, ist OR.

TIMER(datenbereich)

Wenn das angegebene Ereignis ein Zeitgeberereignis ist, gibt diese Option den 16 Zeichen langen Namen des Zeitgebers zurück.

Bedingungen**109 ACTIVITYERR**

RESP2-Werte:

3

Die in der Option ACTIVITYID angegebene Aktivität konnte nicht gefunden werden.

29

Die Repository-Datei ist nicht verfügbar.

30

Es ist ein Ein-/Ausgabefehler für die Repository-Datei aufgetreten.

111 EVENTERR

RESP2-Werte:

1

Das in der Option EVENT angegebene Ereignis konnte nicht gefunden werden.

16 INVREQ

RESP2-Werte:

1

Es ist keine aktuelle Aktivität im Bereich dieser Task vorhanden.

17 IOERR

RESP2-Werte:

30

Es ist ein Ein-/Ausgabefehler für die Repository-Datei aufgetreten.

70 NOTAUTH

RESP2-Werte:

101

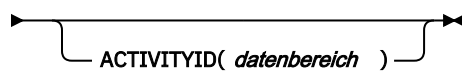
Der Benutzer, der der absetzenden Task zugeordnet ist, ist nicht berechtigt, auf diese Ressource in der angeforderten Weise zuzugreifen.

INQUIRE PROCESS

Ruft die Attribute eines BTS-Prozesses ab.

INQUIRE PROCESS

➡ INQUIRE — PROCESS(*datenwert*) — PROCESSTYPE(*datenwert*) →



Bedingungen: ILLOGIC, NOTAUTH, PROCESSERR

Beschreibung

Der Befehl INQUIRE PROCESS gibt die Attribute eines angegebenen BTS-Prozesses zurück. Er kann beispielsweise dazu verwendet werden, die ID der Stammaktivität eines Prozesses abzurufen, um ein Durchsuchen (Browsing) der untergeordneten Aktivitäten, Container und Ereignisse der Stammaktivität zu starten.

Optionen

ACTIVITYID(datenbereich)

Gibt die 52 Zeichen lange ID der Stammaktivität des Prozesses zurück, der abgefragt wird.

PROCESS(datenwert)

Gibt den Namen (1 - 36 Zeichen) des Prozesses an, der abgefragt werden soll.

PROCESSTYPE(datenwert)

Gibt den Prozesstyp (1 - 8 Zeichen) des Prozesses an, der abgefragt werden soll.

Bedingungen

21 ILLOGIC

RESP2-Werte:

1

Ein Durchsuchen (Browsing) dieses Ressourcentyps ist bereits in Bearbeitung.

70 NOTAUTH

RESP2-Werte:

101

Der Benutzer, der der absetzenden Task zugeordnet ist, ist nicht berechtigt, auf diese Ressource in der angeforderten Weise zuzugreifen.

108 PROCESSERR

RESP2-Werte:

1

Der in der Option PROCESS angegebene Prozess konnte nicht gefunden werden.

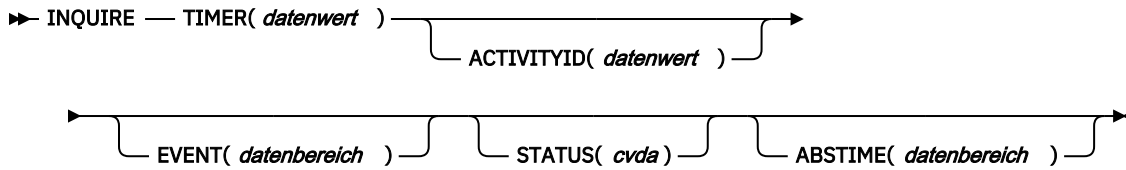
4

Der in der Option PROCESSTYPE angegebene Prozesstyp konnte nicht gefunden werden.

INQUIRE TIMER

Ruft die Attribute eines BTS-Zeitgebers ab.

INQUIRE TIMER



Bedingungen: ACTIVITYERR, INVREQ, IOERR, NOTAUTH, TIMERERR

Beschreibung

Der Befehl **INQUIRE TIMER** gibt die Attribute eines angegebenen BTS-Zeitgebers zurück.

Zum Abfragen eines Zeitgebers, der der aktuellen Aktivität zugeordnet ist, lassen Sie die Option ACTIVITYID weg. Zum Abfragen eines Zeitgebers, der einer anderen Aktivität zugeordnet ist, geben Sie die Option ACTIVITYID an. (Die Aktivitäts-ID, die in der Option ACTIVITYID angegeben wird, kann zum Beispiel durch einen Befehl **GETNEXT ACTIVITY** während einer Browseoperation zurückgegeben worden sein.)

Optionen

ABSTIME(datenbereich)

Gibt den Zeitpunkt, zu dem der Zeitgeber abläuft, im gepackten Dezimalformat in Millisekunden seit dem 1. Januar 1900, 00:00 Uhr, (auf die nächste Hundertstelsekunde aufgerundet) zurück.

Mit dem Befehl **FORMATTIME** können Sie die Daten in andere gängige Formate ändern.

ACTIVITYID(datenwert)

Gibt die ID (1 - 52 Zeichen) der Aktivität an, der der Zeitgeber zugeordnet ist.

Wenn diese Option nicht angegeben wird, wird die aktuelle Aktivität angenommen.

EVENT(datenbereich)

Gibt den 16 Zeichen langen Namen des Ereignisses (sofern vorhanden) zurück, das dem Zeitgeber zugeordnet ist.

STATUS(cvda)

Gibt den Status des Zeitgebers an. Die CVDA-Werte lauten wie folgt:

EXPIRED

Der Zeitgeber ist normal abgelaufen.

FORCED

Der Ablauf des Zeitgebers wurde durch einen Befehl **FORCE TIMER** erzwungen.

UNEXPIRED

Der Zeitgeber ist noch nicht abgelaufen.

TIMER(datenwert)

Gibt den Namen (1 - 16 Zeichen) des Zeitgebers an.

Bedingungen

109 ACTIVITYERR

RESP2-Werte:

3

Die in der Option ACTIVITYID angegebene Aktivität konnte nicht gefunden werden.

29

Die Repository-Datei ist nicht verfügbar.

30

Es ist ein Ein-/Ausgabefehler für die Repository-Datei aufgetreten.

16 INVREQ

RESP2-Werte:

1

Der Befehl wurde außerhalb des Gültigkeitsbereichs einer momentan aktiven Aktivität abgesetzt.

17 IOERR

RESP2-Werte:

30

Es ist ein Ein-/Ausgabefehler für die Repository-Datei aufgetreten.

70 NOTAUTH

RESP2-Werte:

101

Der Benutzer, der der absetzenden Task zugeordnet ist, ist nicht berechtigt, auf diese Ressource in der angeforderten Weise zuzugreifen.

115 TIMERERR

RESP2-Werte:

1

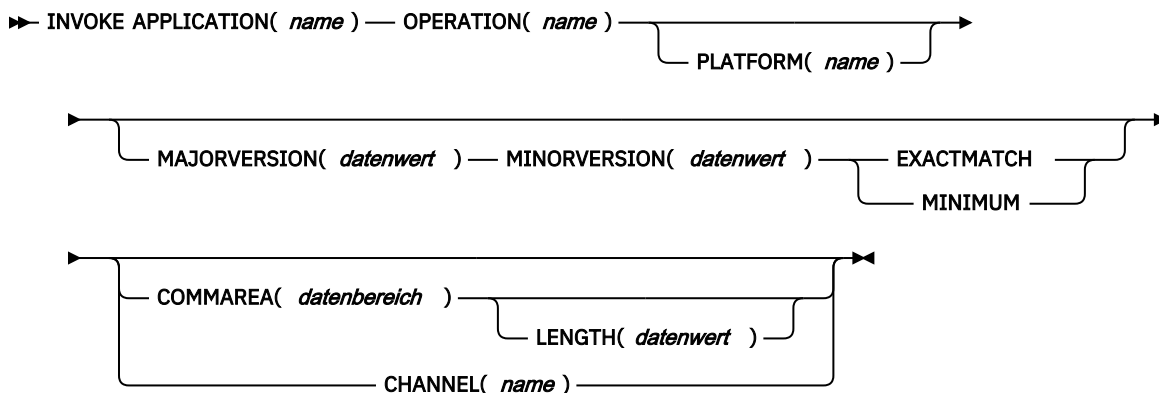
Der in der Option TIMER angegebene Zeitgeber konnte nicht gefunden werden.

INVOKE APPLICATION

Ruft ein Programm an einem Anwendungseinstiegspunkt auf. Der Befehl **EXEC CICS INVOKE APPLICATION** ermöglicht das Aufrufen einer Anwendung durch die Angabe einer Operation, die einem der Anwendungseinstiegspunkte entspricht, ohne den Namen des Programms des Anwendungseinstiegspunkts kennen zu müssen, und unabhängig davon, ob es sich um ein öffentliches oder privates Programm handelt.

Die Anwendung muss sich in einem verfügbaren Status (AVAILABLE) befinden. Wenn mehrere Versionen der Anwendung einen verfügbaren Status haben, sind die Einstiegspunktprogramme der höchsten Version der Anwendung diejenigen, die öffentlich sind, sodass sie durch einen Befehl **EXEC CICS LINK** verknüpft werden können. Der Befehl **INVOKE APPLICATION** lässt das Aufrufen niedrigerer Versionen der Anwendung zu. Dazu muss eine Version und eine Operation angegeben werden. Dies entspricht einem privaten Einstiegspunktprogramm.

INVOKE APPLICATION



Bedingungen: APPNOTFOUND, CHANNELERR, INVREQ, LENGERR, NOTAUTH, PGMIDERR

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl **INVOKE APPLICATION** verknüpft ein lokales Programm, bei dem es sich um den Einstiegs- punkt in die angegebene Anwendung für die angegebene Operation handelt. Er übergibt die Steuerung aus einem Anwendungsprogramm auf einer logischen Ebene an ein Anwendungsprogramm auf der nächstniedrigeren logischen Ebene.

Dieser Befehl operiert im aktuellen Plattformkontext. Wenn der Befehl keinen Plattformnamen angibt, wird der aktuelle Plattformname verwendet. Wenn keine aktuelle Plattform vorhanden ist, schlägt der Befehl mit der Antwort APPNOTFOUND fehl.

Weitere Informationen und Beispiele dazu, wie der Befehl **EXEC CICS INVOKE APPLICATION** verwendet werden kann, finden Sie unter [Anwendung mit mehreren Versionen aufrufen](#).

Optionen

APPLICATION(name)

Gibt den Namen (1 - 64 Zeichen) der Anwendung an. Zulässige Zeichen: a-z A-Z 0-9 . _ # @ -

CHANNEL(name)

Gibt den Namen (1 - 16 Zeichen) eines Kanals an, der für das aufgerufene Programm verfügbar gemacht werden soll. Zulässige Zeichen: A-Z a-z 0-9 \$ @ # / % & ? ! : | " = ~ , ; < > . - _ . Führende und eingebettete Leerzeichen sind nicht zulässig. Wenn der angegebene Name kürzer als 16 Zeichen ist, wird er mit nachfolgenden Leerzeichen bis auf 16 Zeichen aufgefüllt. Wenn der Kanal nicht vorhanden ist, wird er erstellt. Dieser neue Kanal bleibt im Bereich, bis die Verbindungsebene geändert wird. Weitere Informationen zum Kanalbereich finden Sie unter [Geltungsbereich eines Kanals](#).

Kanalnamen haben immer EBCDIC-Format. Der Satz an zulässigen Zeichen für Kanalnamen, wie zuvor aufgeführt, umfasst einige Zeichen, die nicht in allen EBCDIC-Codepages dieselbe Darstellung haben. Wenn Kanäle zwischen Regionen übertragen werden sollen, wird daher empfohlen, die Zeichen zur Kanalbenennung auf die folgenden zu begrenzen: A-Z a-z 0-9 & : = , ; < > - und _.

Sie können den Kanalnamen DFHTRANSACTION angeben, um einen Transaktionskanal zu verwenden. Ein Transaktionskanal verlässt den Bereich nicht, wenn die Verbindungsebene geändert wird: er ist in der Transaktion immer zugänglich. Weitere Informationen finden Sie unter [Kanäle und Container](#).

Für das Programm, das den Befehl INVOKE absetzt, bestehen die folgenden Möglichkeiten:

- Es kann den Kanal durch einen oder mehrere der Befehle **PUT CONTAINER CHANNEL** oder **PUT64 CONTAINER** bereits erstellt haben.
- Es kann seinen aktuellen Kanal durch den Namen angeben.
- Es kann einen Kanal benennen, der zurzeit nicht vorhanden ist. Es wird ein leerer Kanal erstellt.

COMMAREA(datenbereich)

Gibt einen Kommunikationsbereich an, der für das aufgerufene Programm verfügbar gemacht werden soll. In dieser Option wird der Datenbereich übergeben und Sie müssen ihm im empfangenden Programm den Namen DFHCOMMAREA geben. Siehe [Daten an andere Programme übergeben](#).

EXACTMATCH

Gibt an, dass eine exakte Übereinstimmung mit der Hauptversionsnummer und der Nebenversionsnummer der Anwendung erforderlich ist. Wenn die Anwendung nicht gefunden wird, wird die Bedingung APPNOTFOUND zurückgegeben.

Anmerkung: Es gibt kein Übereinstimmungskriterium für die Mikroversion. Es wird immer die höchste Mikroversion verwendet.

LENGTH(datenwert)

Gibt einen Halbwort-Binärwert an, der die Länge des Kommunikationsbereichs (COMMAREA) in Byte darstellt. Dieser Wert darf 24 KB nicht überschreiten, wenn der Kommunikationsbereich zwischen zwei beliebigen CICS-Servern (in jeder Kombination aus Produkt, Version und Release) übergeben werden soll. Diese Begrenzung berücksichtigt den Kommunikationsbereich und den Platz für Header.

Stellen Sie sicher, dass der Wert, den Sie angeben, der Länge der Daten entspricht, die in dem Kommunikationsbereich übergeben werden. Geben Sie für LENGTH nicht 0 (null) an, weil das daraus resultierende Verhalten unvorhersehbar ist und der Befehl **EXEC CICS LINK** fehlschlagen könnte.

Wenn Sie einen Kommunikationsbereich zur Übergabe von Daten verwenden, muss das verknüpfte Programm prüfen, ob das Feld EIBCALEN in dem EIB der Task mit dem Wert übereinstimmt, den das Programm erwartet. Abweichungen könnten zu Fehlern im Speicher oder zu Systemfehlern führen. Weitere Informationen finden Sie unter [COMMAREA](#).

MAJORVERSION(datenwert)

Gibt die Hauptversionsnummer der Anwendung als Vollwort-Binärwert an.

Wenn MAJORVERSION angegeben wird, muss auch MINORVERSION angegeben werden. Wenn keine Version angegeben wird, wird die Anwendung mit der höchsten Haupt- und Nebenversion aufgerufen.

MINIMUM

Gibt an, dass die angegebene Nebenversionsnummer die minimal erforderliche Nummer ist und dass eine höhere Version verwendet werden soll, sofern sie verfügbar ist. Wenn mehrere höhere Nebenversionen verfügbar sind, wird die höchste Nummer verwendet. Dies gilt nur für die Nebenversionsnummer. Die Hauptversionsnummer kann nicht überschritten werden und muss exakt übereinstimmen. Wenn weder eine höhere Nebenversion noch die erforderliche Mindestversion vorhanden ist, wird die Bedingung APPNOTFOUND zurückgegeben.

Anmerkung: Es gibt kein Übereinstimmungskriterium für die Mikroversion. Es wird immer die höchste Mikroversion verwendet.

MINORVERSION(datenwert)

Gibt die Nebenversionsnummer der Anwendung als Vollwort-Binärwert an.

Wenn MINORVERSION angegeben wird, muss auch MAJORVERSION angegeben werden. Wenn keine Version angegeben wird, wird die Anwendung mit der höchsten Haupt- und Nebenversion aufgerufen.

Die Schlüsselwörter EXACTMATCH oder MINIMUM geben die Übereinstimmungskriterien für die Haupt- und Nebenversion an.

OPERATION(name)

Gibt den Namen (1 - 64 Zeichen) der Anwendungsoperation an, die von dem Programm des Anwendungseinstiegspunkts implementiert wird. Zulässige Zeichen: a-z A-Z 0-9 . _ # @ -

PLATFORM(name)

Gibt den Namen (1 - 64 Zeichen) der Plattform an, auf der die Anwendung installiert ist. Zulässige Zeichen: a-z A-Z 0-9 . _ # @ -

Wenn kein Plattformname angegeben wird, wird der aktuelle Plattformname verwendet. Wenn keine aktuelle Plattform vorhanden ist, schlägt der Befehl mit der Antwort APPNOTFOUND fehl.

Bedingungen

127 APPNOTFOUND

RESP2-Werte:

1

Das Schlüsselwort EXACTMATCH wurde angegeben und die erforderliche Version der Anwendung kann nicht gefunden werden.

2

Das Schlüsselwort MINIMUM wurde angegeben und es kann keine Mindestversion oder höhere Mikroversion der Anwendung gefunden werden.

3

Es wurde keine Version angegeben. Es kann keine Anwendung gefunden werden.

Standardaktion: Die Task wird abnormal beendet.

Eine Anwendung kann aus einer Reihe von Gründen nicht gefunden werden:

- Die Anwendung befindet sich nicht in einem verfügbaren Status (AVAILABLE).

- Die benannte Operation entspricht keinem Einstiegspunktprogramm für die Anwendung.
- Die Anwendung ist auf dieser Plattform nicht installiert.
- Die CICS-Region ist kein Teil des angegebenen Plattformnamens.

122 CHANNELERR

RESP2-Werte:

1

Der Name, der in der Option CHANNEL angegeben wurde, enthält ein ungültiges Zeichen oder eine ungültige Kombination von Zeichen.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

RESP2-Werte:

1

Es ist kein Plattformname angegeben und es ist keine aktuelle Plattform vorhanden.

2

Das Anwendungseinstiegspunktprogramm ist ein Java™-Programm, jedoch kann die Benutzerklasse nicht gefunden werden.

3

Das Anwendungseinstiegspunktprogramm ist ein Java-Programm, jedoch kann der JVM-Server (JVMSEVER) nicht gefunden werden.

4

Das Anwendungseinstiegspunktprogramm ist ein Java-Programm, jedoch ist die JVMSEVER-Ressource nicht aktiviert.

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

RESP2-Werte:

11

Die Länge von COMMAREA ist kleiner als 0 oder größer als die zulässige Länge.

26

Die Adresse von COMMAREA ist null, jedoch ist die Länge von COMMAREA ungleich null.

Standardaktion: Die Task wird abnormal beendet.

70 NOTAUTH

RESP2-Werte:

101

Eine Ressourcensicherheitsprüfung ist für den Namen des Anwendungseinstiegspunktprogramms fehlgeschlagen, das die Operation für die angegebene Operation implementiert.

Standardaktion: Die Task wird abnormal beendet.

27 PGMIDERR

RESP2-Werte:

1

Das Anwendungseinstiegspunktprogramm ist inaktiviert.

2

Das Anwendungseinstiegspunktprogramm konnte nicht geladen werden.

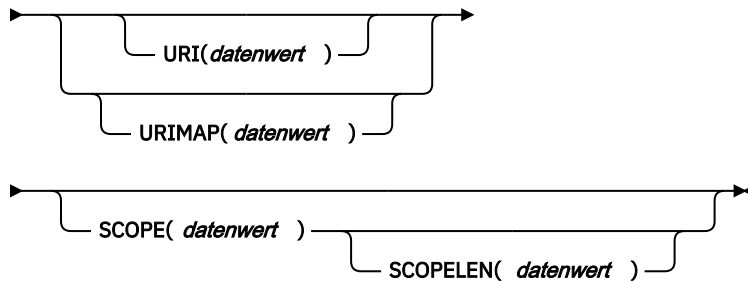
Standardaktion: Die Task wird abnormal beendet.

INVOKE SERVICE

Ruft einen Service über eine CICS-Anwendung auf. Der Befehl gibt den Namen eines Service oder der CICS-Ressource an, wie zum Beispiel eine WEBSERVICE-Ressource, die Informationen zu dem aufzurufenden Service enthält.

INVOKE SERVICE

➔ INVOKE SERVICE(*datenwert*) — CHANNEL(*datenwert*) — OPERATION(*datenwert*) ➔



Bedingungen: INVREQ, LENGERR, NOTFND, TIMEDOUT

Dieser Befehl ist threadsicher.

Beschreibung

Verwenden Sie den Befehl **INVOKE SERVICE** in CICS-Anwendungen, um einen Service aufzurufen. Zum Beispiel kann die Anwendung als Web-Service-Requester fungieren und einen XML-basierten Service aufrufen oder sie kann eine andere CICS-Anwendung aufrufen, die als kanalbasierter Service fungiert. Weitere Informationen zu diesen beiden Typen von Services finden Sie unter [Geschäftsservices aus CICS-Anwendungen erstellen](#).

Verwenden Sie diesen Befehl für alle neuen Web-Service-Anwendungen anstelle des Befehls **INVOKE WEBSERVICE**, der ein Synonym ist. Wenn Sie diesen Befehl für Web-Services verwenden, müssen Sie bestimmte Container für CICS als Eingabe angeben. Weitere Informationen zum Schreiben einer Web-Service-Anwendung finden Sie unter [SOAP-Web-Service erstellen](#).

Wenn Sie den Service aufrufen, können Sie eine URIMAP-Ressource angeben, die die Informationen zum URI des Service enthält. Sie können diese Informationen direkt im Befehl **INVOKE SERVICE** angeben, ohne eine URIMAP-Ressource zu verwenden. Die Verwendung einer URIMAP-Ressource bietet allerdings die folgenden Vorteile:

- Systemadministratoren können alle Änderungen am Endpunkt der Verbindung verwalten, sodass Sie Ihre Anwendungen nicht erneut kompilieren müssen, wenn sich der URI eines Service-Providers ändert.
- Sie können CICS veranlassen, die Verbindungen, die mit der URIMAP-Ressource geöffnet wurden, nach der Verbindung geöffnet zu halten und in einen Pool zu stellen, sodass sie von der Anwendung für nachfolgende Anforderungen oder von einer anderen Anwendung, die denselben Service aufruft, wiederverwendet werden können. Das Verbindungspooling ist nur verfügbar, wenn Sie eine URIMAP-Ressource angeben, für die das Attribut SOCKETCLOSE festgelegt ist. Weitere Informationen zu den Leistungsvorteilen, die sich durch das Verbindungspooling ergeben, finden Sie unter [Verbindungspooling für HTTP-Clientleistung](#).

Der Befehl **INVOKE SERVICE** nutzt den Benutzerexit XWBOPEN, mit dessen Hilfe die Verbindung zu dem Server bei Bedarf durch einen Proxy-Server geleitet werden kann.

Optionen

CHANNEL(*datenwert*)

Gibt den Namen des Kanals an, der zur Übergabe der Container verwendet wird, die die Daten enthalten, die durch die Anwendungsdatenstruktur zugeordnet werden. Nach der Rückgabe enthält derselbe Kanal die Antwort aus dem Web-Service, die wieder durch die Anwendungsdatenstruktur zugeordnet wird. Der Name des Kanals kann bis zu 16 Zeichen lang sein. Wenn *name* eine Variable ist, die

einen Namen enthält, der weniger als 16 Zeichen lang ist, muss die Variable mit nachfolgenden Leerzeichen aufgefüllt werden. Sie können den Kanalnamen DFHTRANSACTION angeben, um den Transaktionskanal zu verwenden.

OPERATION (datenwert)

Gibt einen Datenbereich an, der den Namen der Operation enthält, die aufgerufen werden soll. Der Name der Operation ist in der WSDL-Datei für den Ziel-Web-Service enthalten. Der Datenbereich muss 255 Zeichen lang sein. Wenn der Operationsname kürzer als 255 Zeichen ist, muss der Datenbereich mit nachfolgenden Leerzeichen aufgefüllt werden.

SERVICE (datenwert)

Gibt den Namen des Service an.

- Wenn Sie einen Web-Service aufrufen wollen, geben Sie den Namen der WEBSERVICE-Ressource an, die den Web-Service definiert. Die WEBSERVICE-Ressource gibt die Position der Web-Service-Beschreibung und der Web-Service-Bindungsdatei an, die von CICS für die Kommunikation mit dem Web-Service verwendet werden. Der Name der WEBSERVICE-Ressource kann bis zu 32 Zeichen lang sein. Der Wert muss mit nachfolgenden Leerzeichen aufgefüllt werden, wenn er weniger als 32 Zeichen enthält.
- Wenn Sie einen kanalbasierten Service aufrufen wollen, geben Sie den Namen des Service an. Das Format des Service ist ein URI. Der Name kann bis zu 32 Zeichen lang sein. Der Wert muss mit nachfolgenden Leerzeichen aufgefüllt werden, wenn er weniger als 32 Zeichen enthält.

SCOPE (datenwert)

Fungiert als qualifizierendes Präfix für den Servicenamen. Verwenden Sie die Option SCOPE, wenn Sie die kanalbasierten SCA-Services (SCA - Service Component Architecture) verwenden wollen und ein zusätzliches Präfix zur Identifikation des Service benötigen.

Anmerkung: SCOPE gilt nur für kanalbasierte Services.

Wenn Sie ein COBOL-Programm schreiben, das mit der Umsetzeroption COBOL3 umgesetzt wird, darf die Länge des Datenwerts 160 Byte nicht überschreiten. Wenn Sie die Umsetzeroption COBOL2 verwenden, müssen Sie anstelle eines Datenwerts einen Datenbereich verwenden.

SCOPELEN (datenwert)

Ein Vollwort-Binärwert, der die Länge des Werts in SCOPE angibt, der als Präfix für den Servicenamen verwendet wird.

URI (datenwert)

Gibt einen Datenbereich an, der den URI des aufzurufenden Service enthält. Wenn diese Option angegeben wird, ersetzt Sie jeden URI, der in der WEBSERVICE-Ressourcendefinition angegeben ist. Wenn Sie weder diese Option noch die Option URIMAP angeben, muss die WEBSERVICE-Bindungsdatei, die der Ressourcendefinition zugeordnet ist, entweder einen Provider-URI oder einen Provideranwendungsnamen enthalten. Der Datenbereich muss 255 Zeichen lang sein. Wenn der URI kürzer als 255 Zeichen ist, muss der Datenbereich mit nachfolgenden Leerzeichen aufgefüllt werden. Informationen zum Format von URIs finden Sie unter [Komponenten einer URL](#).

Geben Sie diese Option nicht für Web-Service an, die WS-Addressing verwenden.

Geben Sie diese Option nicht an, wenn Sie das Verbindungspooling verwenden. Verwenden Sie stattdessen die Option URIMAP, um eine entsprechende URIMAP-Ressource zur Aktivierung des Verbindungspoolings anzugeben.

URIMAP (datenwert)

Gibt den Namen einer URIMAP-Ressource an, die von CICS zum Ableiten des URI-Werts verwendet wird. Verwenden Sie eine URIMAP-Ressource, wenn Sie das Verbindungspooling aktivieren wollen, bei dem CICS die HTTP-Clientverbindung zur Wiederverwendung durch diese Anwendung oder eine andere Anwendung geöffnet hält. Wenn diese Option angegeben wird, ersetzt Sie jeden URI, der in der WEBSERVICE-Ressourcendefinition angegeben ist. Wenn Sie weder diese Option noch die Option URI angeben, muss die WEBSERVICE-Bindungsdatei, die der Ressourcendefinition zugeordnet ist, entweder einen Provider-URI oder einen Provideranwendungsnamen enthalten.

Sie müssen die URIMAP-Ressource für eine HTTP-Clientanforderung mit dem Attribut USAGE(CLIENT) erstellen. Für das Verbindungspooling müssen Sie außerdem das Attribut SOCKETCLOSE festlegen. Der CICS-Web-Service-Assistent erstellt die URIMAP-Ressource nicht, sodass Sie sie selbst defi-

nieren müssen. Informationen zur Erstellung einer URIMAP-Ressource für eine Clientanforderung finden Sie unter [URIMAP-Ressource für CICS als HTTP-Client erstellen](#).

Geben Sie diese Option nicht für Web-Service an, die WS-Addressing verwenden.

Bedingungen

16 INVREQ

RESP2-Werte:

- 1**
Der angegebene Name für die Option CHANNEL enthält ein ungültiges Zeichen oder eine ungültige Kombination von Zeichen.
- 2**
Der angegebene Name für die Option OPERATION enthält ein ungültiges Zeichen oder eine ungültige Kombination von Zeichen.
- 3**
Die Web-Service-Bindungsdatei, die dem Web-Service (WEBSERVICE) zugeordnet ist, ist ungültig.
- 4**
Der angegebene Wert für den URI enthielt ein unzulässiges Zeichen oder eine unzulässige Kombination von Zeichen oder der angegebene Hostname konnte nicht aufgelöst werden.
- 5**
Die Pipeline, die von dem Web-Service verwendet wird, ist als Service-Requester-Pipeline definiert, jedoch wird sie in einem Service-Provider aufgerufen oder *umgekehrt*.
- 6**
Der aufgerufene Web-Service hat einen SOAP-Fehler zurückgegeben. Die Beschreibung des Fehlers ist im XML-Format im Container DFHWS-BODY verfügbar.
Anmerkung: Diese Bedingung wird nicht für XML-ONLY-Aufrufe von Web-Services ausgelöst.
- 7**
Die Option URI wurde im Befehl nicht angegeben und die WEBSERVICE-Definition gibt weder einen URI oder noch einen Programmnamen an.
- 8**
Der Web-Service ist nicht in Betrieb.
- 9**
Ein Container hat nicht den korrekten Datentyp (DATATYPE). Dies kann der Container **DFHWS-DATA** oder ein anderer Container sein, auf den in den Anwendungsdaten verwiesen wird. Der Container **DFHWS-DATA** und die meisten anderen Container für Anwendungsdaten müssen im BIT-Modus gefüllt sein. Alle Container, die XML-Markup enthalten, müssen im CHAR-Modus gefüllt sein.
- 10**
Die Pipeline, die von dem Web-Service verwendet wird, ist nicht aktiviert.
- 11**
CICS konnte keine Verbindung zu dem Programm herstellen, das in der WEBSERVICE-Definition angegeben ist.
- 12**
Die Container, die der Befehl erwartet, waren nicht im richtigen Kanal enthalten.
- 13**
Ein Eingabefehler wurde entweder bei der Generierung einer SOAP-Anforderungsnachricht oder bei der Verarbeitung einer SOAP-Antwortnachricht festgestellt. Eine Nachricht DFHPIxxxx wurde in MSGUSR geschrieben, um das Problem detaillierter zu dokumentieren. Es ist wahrscheinlich, dass die Anwendungsdatenstruktur ungültige Daten enthält, die nicht in eine SOAP-Anforderungsnachricht umgewandelt werden können. Weitere Informationen finden Sie in der Fehlernachricht im Container DFH-XML-ERRORMSG.

14

Es ist ein Konvertierungsfehler aufgetreten, als CICS versucht hat, eine Konvertierung zwischen der Anwendungsdatenstruktur und der SOAP-Nachricht durchzuführen. Entweder enthält die Anwendungsdatenstruktur ungültige Daten, die nicht in eine SOAP-Anforderung umgewandelt werden können, oder Daten in der SOAP-Antwortnachricht können nicht in die Datenstruktur der Anwendung umgewandelt werden. Diese Bedingung kann zum Beispiel die folgenden Ursachen haben:

- Ein Wert in der SOAP-Antwortnachricht ist größer als das entsprechende Feld in der Datenstruktur der Anwendung.
- Bei der Erstellung der SOAP-Anforderung gibt die Web-Service-Bindungsdatei an, dass ein Datenfeld gepackte Dezimaldaten oder gezonte Dezimaldaten enthält, und der Inhalt des Felds ist für diesen Datentyp ungültig.

Eine Nachricht DFHPIxxxx wurde in MSGUSR geschrieben, um das Problem detaillierter zu dokumentieren. Weitere Informationen finden Sie in der Fehlernachricht im Container DFH-XML-ERRORMSG.

15

Es ist ein nicht behandelter Fehler in der Pipeline aufgetreten. Informationen zu dem Fehler befinden sich im Container DFHERROR.

16

Ein lokal optimierter Web-Service wurde abnormal beendet (Abbruch). Die zugrunde liegende Arbeitseinheit (UOW) wurde zurückgesetzt.

17

Eine ferne Web-Service-Anforderung hat keine Antwortnachricht zurückgegeben.

18

Der Container **DFHWS - BODY** wurde nicht durch eine Anwendung für einen XML-ONLY-Web-Service gefüllt.

19

Eine Option URI oder URIMAP wurde angegeben, jedoch ist diese Option nicht zulässig, wenn die WEBSERVICE-Ressource eine standardmäßig definierte WS-Addressing-Endpunktreferenz besitzt oder wenn der WS-Addressing-Kontext mit dem API-Befehl **WSACONTEXT BUILD** erstellt wurde.

20

Die angegebene URIMAP-Ressource hat kein gültiges Schema.

21

Die angegebene URIMAP-Ressource ist nicht im Clientmodus.

22

Die angegebene URIMAP-Ressource ist nicht aktiviert.

23

Ein nicht angegebener Transport- oder Verbindungsfehler ist bei dem Versuch aufgetreten, die Pipeline zu verwenden. CICS gibt eine Nachricht aus, um das betreffende Problem zu dokumentieren.

24

z/OS Connect-Pipeline wurde verwendet.

41

Die Verbindung wurde geschlossen.

101

Der Container **DFHWS - BODY** hat nicht den richtigen Datentyp (DATATYPE). Für diesen Container muss für DATATYPE der Datentyp CHAR angegeben werden.

103

Der Container **DFHWS - BODY** enthält keine Daten.

104

Entweder der Container **DFHREQUEST** oder der Container **DFHWS - BODY** fehlt.

105

Ein Fehler wurde in der Service-Requester-Pipeline, die von dem Web-Service verwendet wird, entweder beim Senden der Anforderung oder beim Verarbeiten der Antwort erstellt. Diese Bedingung könnte darauf hinweisen, dass ein Programm zur Headerverarbeitung einen Fehler ausgegeben hat.

106

Entweder war die generierte SOAP-Anforderungsnachricht nicht ordnungsgemäß formatiert oder die SOAP-Antwortnachricht war nicht ordnungsgemäß formatiert. Diese Bedingung könnte darauf hinweisen, dass der XML-Parser einen schwerwiegenden Fehlercode zurückgegeben hat.

107

Entweder war die generierte SOAP-Anforderungsnachricht keine gültige SOAP-Nachricht oder die SOAP-Antwortnachricht war keine gültige SOAP-Nachricht formatiert.

108

Ein HTTP-Fehler ist aufgetreten. Verwenden Sie den im Container DFHRESPONSE verfügbaren HTTP-Statuscode, um die Ursache dieses HTTP-Fehlers zu diagnostizieren.

109

Es wurde eine HTTP-Umleitungsantwort zum Code 109 (301, 302, 303 oder 307) zurückgegeben. Der Standortheader (Location Header) ist im Container **DFHWS-LOCATION** verfügbar.

22 LENGERR

Werte für RESP:

1

Entweder wurde die Option SCOPELEN nicht angegeben oder sie enthält keinen gültigen Wert.

13 NOTFND

RESP2-Werte:

1

Die Web-Service-Bindungsdatei, die dem Web-Service zugeordnet ist, gibt den Namen eines Parsing-Programms für SOAP-Nachrichten an, das durch ein anderes Produkt bereitgestellt wird, jedoch konnte das Parsing-Programm nicht gefunden werden.

2

Der angegebene Kanal (CHANNEL) konnte nicht gefunden werden.

3

Die angegebene Operation befand sich nicht in der Web-Service-Bindungsdatei.

4

Der angegebene Web-Service konnte nicht gefunden werden.

5

Ein in der Web-Service-Bindungsdatei angegebener Container konnte nicht gefunden werden.

6

Die angegebene URIMAP-Ressource konnte nicht gefunden werden.

124 TIMEDOUT

RESP2-Werte:

1

Es ist eine erwartete Zeitlimitüberschreitung aufgetreten. Wenn das Nachrichtenaustauschmuster eine optionale Fehlerantwort angibt und von dem fernen Web-Service keine Fehlerantwort zurückgegeben wird, ist die Zeitlimitüberschreitung annehmbar.

2

Es ist eine unerwartete Zeitlimitüberschreitung aufgetreten. Es wurde eine Antwort von dem fernen Web-Service erwartet, jedoch keine empfangen.

62

Es ist eine unerwartete Zeitlimitüberschreitung beim Socketempfang aufgetreten.

INVOKE WEBSERVICE

Ruft einen Service über eine CICS-Anwendung auf. Dieser Befehl ist ein Synonym für den Befehl **INVOKE SERVICE** und wird aus Gründen der Kompatibilität mit bestehenden Web-Service-Requester-Anwendungen bereitgestellt. Verwenden Sie den Befehl **INVOKE SERVICE** für alle neuen Web-Service-Anwendungen.

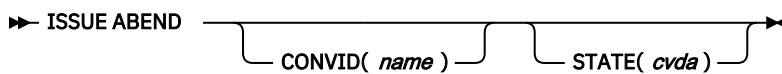
Beschreibung

Informationen zum Befehl **INVOKE SERVICE** finden Sie unter „[INVOKE SERVICE](#)“ auf Seite 282.

ISSUE ABEND

Bewirkt eine abnormale Beendigung (Abbruch) des formatfreien Dialogs mit einem APPC-Partner.

ISSUE ABEND (APPC)



Bedingungen: INVREQ, NOTALLOC, TERMERR

Beschreibung

Der Befehl **ISSUE ABEND** beendet den Dialog abnormal (Abbruch). Die Partnertransaktion empfängt die Bedingung **TERMERR**.

Optionen

CONVID(name)

Gibt den Dialog an, der abnormal beendet werden soll. Der vier Zeichen lange Name bezeichnet entweder die symbolische ID, die von einem zuvor ausgeführten Befehl **ALLOCATE** in **EIBRSRCE** im **EIB** zurückgegeben wurde, oder die symbolische ID, die die Hauptfunktion (die durch einen zuvor ausgeführten Befehl **ASSIGN** zurückgegeben wurde) darstellt.

Aus Gründen der Kompatibilität mit früheren Releases wird **SESSION** als Synonym für **CONVID** akzeptiert. Neue Programme sollten **CONVID** verwenden.

Wenn weder **CONVID** noch **SESSION** angegeben wird, wird die Hauptfunktion angenommen.

STATE(cvda)

Ruft den Status des aktuellen Datenaustausches ab. Die folgenden **CVDA**-Werte können von **CICS** zurückgegeben werden:

- **ALLOCATED**
- **CONFFREE**
- **CONFRECEIVE**
- **CONFSEND**
- **FREE**
- **PENDFREE**
- **PENDRECEIVE**
- **RECEIVE**
- **ROLLBACK**
- **SEND**

- SYNCFREE
- SYNCRECEIVE
- SYNCSEND

Bedingungen

16 INVREQ

RESP2-Werte:

200

Eine Serveranwendung für Distributed Program Link hat in der Option CONVID die Sitzung für die Funktionsverlagerung (ihre Hauptfunktion) angegeben.

Tritt außerdem in jedem der folgenden Fälle auf (RESP2 nicht festgelegt):

- Der Befehl ISSUE ABEND wird für einen anderen Dialog als einen formatfreien EXEC CICS APPC-Dialog verwendet.

Standardaktion: Die Task wird abnormal beendet.

61 NOTALLOC

Tritt auf, wenn sich der in CONVID angegebene Wert auf einen Dialog bezieht, der nicht der Anwendung gehört.

Standardaktion: Die Task wird abnormal beendet.

81 TERMERR

Tritt für einen sitzungsbezogenen Fehler auf. Jede Aktion außer einem Befehl FREE für diesen Dialog verursacht einen Abbruch mit dem Code ATCV.

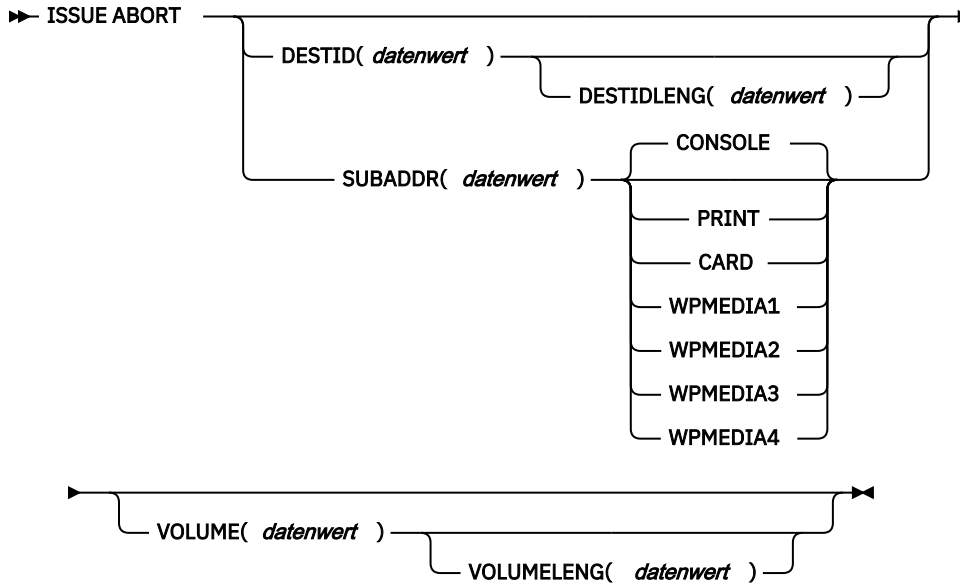
Eine Anforderung CANCEL TASK durch ein Benutzerknotenfehlerprogramm (NEP) kann diese Bedingung auslösen, wenn die Task eine ausstehende aktive Terminalsteueranforderung hat, wenn das Programm für abnormale Knotenbedingungen den Sitzungsfehler verarbeitet.

Standardaktion: Die Task wird abnormal mit dem Abbruchcode ATNI beendet.

ISSUE ABORT

Beendet die Verarbeitung einer Datei (Dataset) abnormal.

ISSUE ABORT



Bedingungen: FUNCERR, INVREQ, SELNERR, UNEXPIN

Beschreibung

Der Befehl **ISSUE ABORT** beendet die Kommunikation mit einer Datei (Dataset) in einem externen Controller oder in dem ausgewählten Medium abnormal. Die Datei, die in der Option **DESTID** angegeben wird, wird abnormal abgewählt. Die Optionen **CONSOLE**, **PRINT**, **CARD** und **WPMEDIA1–4** sind Alternativen für **DESTID** und **DESTIDLENG**.

Optionen

CARD

Gibt an, dass das Ausgabemedium ein Kartenlese- oder Kartenstanzgerät ist. Diese Option ist mit den Optionen **DESTID** und **DESTIDLENG** nicht gültig.

CONSOLE

Gibt an, dass das Ausgabemedium dasjenige ist, das für Nachrichten an den Bediener vorgesehen ist. Diese Option ist mit den Optionen **DESTID** und **DESTIDLENG** nicht gültig. Diese Option bezieht sich auf ein programmierbares Subsystem wie das Datenübertragungssystem IBM 3790. Sie bezieht sich nicht auf eine CICS- oder Systemkonsole.

DESTID(datenwert)

Gibt den Namen (1 - 8 Zeichen) der Datei in dem externen Ziel an.

DESTIDLENG(datenwert)

Gibt die Länge (Halbwort-Binärwert) des Namens an, der in der Option **DESTID** angegeben wird.

PRINT

Gibt an, dass das Ausgabemedium ein Drucker ist.

SUBADDR(datenwert)

Gibt die Unteradresse des Mediums als Halbwort-Binärwert (im Bereich von 0 bis 15) an, die es ermöglicht, Medien desselben Typs wie zum Beispiel "printer 1" oder "printer 2" zu definieren. Der Wert 15 bedeutet ein Medium eines beliebigen Typs. Der Standardwert ist null.

VOLUME(datenwert)

Gibt den Namen (1 - 6 Zeichen) einer Diskette in einem externen Ziel an, die die Datei enthält, die in der Option **DESTID** angegeben wird.

VOLUMELENG(*datenwert*)

Gibt die Länge (Halbwort-Binärwert) des Namens an, der in der Option VOLUME angegeben wird.

WPMEDIA1 bis WPMEDIA4

Gibt für jede bestimmte Einheit vom LU-Typ 4 (LUTYPE4) an, dass ein Textverarbeitungsmedium in Bezug auf eine bestimmte Ein-/Ausgabeeinheit definiert ist.

Bedingungen**48 FUNCERR**

Wird ausgelöst, wenn ein Fehler bei der Ausführung des Befehls auftritt. Die Zielauswahl bleibt davon unbeeinträchtigt und andere Befehle für dasselbe Ziel werden möglicherweise erfolgreich ausgeführt.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

RESP2-Werte:

200

Eine Serveranwendung für Distributed Program Link hat in der Option CONVID die Sitzung für die Funktionsverlagerung (ihre Hauptfunktion) angegeben.

Standardaktion: Die Task wird abnormal beendet.

47 SELNERR

Wird ausgelöst, wenn ein Fehler bei der Zielauswahl auftritt. Das Ziel wird nicht ausgewählt und andere Befehle für dasselbe Ziel werden wahrscheinlich nicht erfolgreich ausgeführt.

Standardaktion: Die Task wird abnormal beendet.

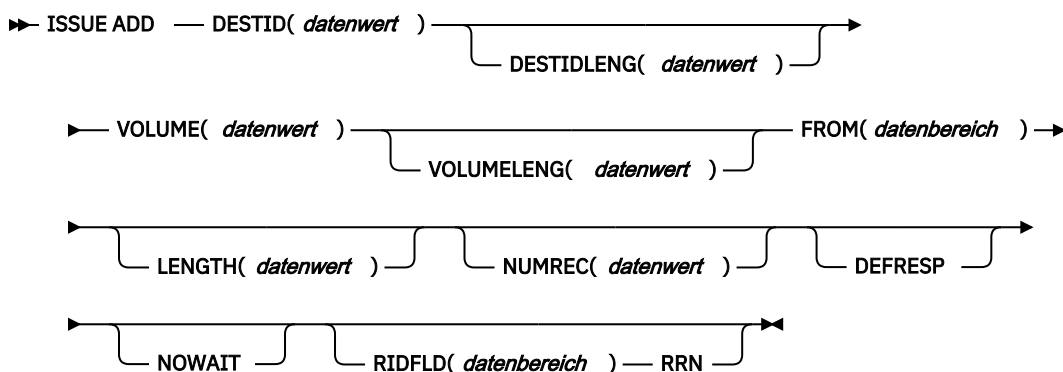
49 UNEXPIN

Tritt auf, wenn nicht erwartete oder nicht erkannte Informationen vom externen Controller empfangen werden.

Standardaktion: Die Task wird abnormal beendet.

ISSUE ADD

Fügt einer Datei (Dataset) einen Datensatz hinzu.

ISSUE ADD

Bedingungen: FUNCERR, INVREQ, SELNERR, UNEXPIN

Beschreibung

Der Befehl **ISSUE ADD** fügt Datensätze einer sequenziellen Datei oder einer direkt organisierten Datei mit Schlüssel in einem externen Controller hinzu. Die Option **FROM** wird zur Angabe der zu schreibenden Daten verwendet und die Option **LENGTH** gibt die Länge der Daten an.

Die Option **RIDFLD** wird für diesen Befehl nur benötigt, wenn er sich auf eine **DPCX/DXAM**-Datei bezieht. In diesem Fall gibt sie die relative Satznummer des hinzuzufügenden Datensatzes an. Wenn **RIDFLD** verwendet wird, muss **NUMREC** den Wert 1 (Standardwert) haben.

Optionen

DEFRESP

Gibt an, dass alle Terminalsteuerungsbefehle, die infolge des Befehls **ISSUE ADD** abgesetzt werden, eine unbedingte Antwort vom externen Stapelverarbeitungsprogramm anfordern sollen, und dies unabhängig von der Angabe der Nachrichtenintegrität für die CICS-Task (durch den Systemprogrammierer).

DESTID(datenwert)

Gibt den Namen (1 - 8 Zeichen) der Datei in dem externen Ziel an.

DESTIDLENG(datenwert)

Gibt die Länge (Halbwort-Binärwert) des Namens an, der in der Option **DESTID** angegeben wird.

FROM(datenbereich)

Gibt die Daten an, die in die Datei zu schreiben sind.

LENGTH(datenwert)

Gibt die Länge (Halbwort-Binärwert) der zu schreibenden Daten an. Eine Beschreibung einer sicheren Obergrenze finden Sie unter [„LENGTH-Optionen in CICS-Befehlen“](#) auf Seite 10.

NOWAIT

Gibt an, dass die CICS-Task die Verarbeitung fortsetzt, ohne auf den Abschluss des Befehls **ISSUE ADD** zu warten. Wenn diese Option nicht angegeben wird, wird die Taskaktivität bis zum Abschluss des Befehls ausgesetzt.

NUMREC(datenwert)

Gibt für eine Datei für relative Sätze (**RRD** - Relative Record Dataset) einen Halbwort-Binärwert für die Anzahl der logischen Datensätze an, die hinzugefügt werden sollen. Datensätze werden sequenziell ab dem Datensatz ersetzt, der durch die Option **RIDFLD** angegeben wird.

Für eine indexierte Datei kann **NUMREC** nicht angegeben werden, da nur ein Datensatz hinzugefügt werden kann.

RIDFLD(datenbereich)

Gibt für eine Datei für relative Sätze ein vier Zeichen langes Feld für die relative Satznummer (beginnend bei null) des Datensatzes an. Die Option **RRN** ist ebenfalls erforderlich.

Für eine direkt organisierte Datei mit Schlüssel muss die Option **RIDFLD** einen Schlüssel angeben.

RRN

Gibt an, dass das Datensatz-ID-Feld, das in der Option **RIDFLD** angegeben wird, eine relative Satznummer (**RRN** - Relative Record Number) enthält. Diese Option ist für eine Datei für relative Sätze erforderlich.

VOLUME(datenwert)

Gibt den Namen (1 - 6 Zeichen) einer Diskette in einem externen Ziel an, die die Datei enthält, die in der Option **DESTID** angegeben wird.

VOLUMELENG(datenwert)

Gibt die Länge (Halbwort-Binärwert) des Namens an, der in der Option **VOLUME** angegeben wird.

Bedingungen

Tritt auf, wenn ein Fehler bei der Ausführung des Befehls festgestellt wird. Die Zielauswahl bleibt davon unbeeinträchtigt und andere Befehle für dasselbe Ziel werden möglicherweise erfolgreich ausgeführt.

Standardaktion: Die Task wird abnormal beendet.

Eine Serveranwendung für Distributed Program Link hat in der Option CONVID die Sitzung für die Funktionsverlagerung (ihre Hauptfunktion) angegeben.

Standardaktion: Die Task wird abnormal beendet.

Standardaktion: Die Task wird abnormal beendet.

Setzt eine positive Antwort auf einen Befehl SEND CONFIRM in einem formatfreien APPC-Dialog ab.

292 CICS TS for z/OS: API-Referenz

- CONFRECEIVE
- CONFSEND
- FREE
- PENDFREE
- PENDRECEIVE
- RECEIVE
- ROLLBACK
- SEND
- SYNCFREE
- SYNCRECEIVE
- SYNCSEND

Bedingungen

16 INVREQ

RESP2-Werte:

200

Eine Distributed Program Link-Serveranwendung hat die Funktionsverlagerungssitzung in der Option CONVID angegeben.

Tritt außerdem in jedem der folgenden Fälle auf (RESP2 nicht festgelegt):

- Der Befehl ISSUE CONFIRMATION wird in einem Dialog verwendet, für den die folgenden Bedingungen zutreffen:
 - Synchronisationsebene 0
 - Kein formatfreier APPC-Dialog

Standardaktion: Die Task wird abnormal beendet.

61 NOTALLOC

Tritt auf, wenn sich der in CONVID angegebene Wert auf einen Dialog bezieht, der nicht der Anwendung gehört.

Standardaktion: Die Task wird abnormal beendet.

24 SIGNAL

Tritt auf, wenn ein eingehender Datenflusssteuerbefehl SIGNAL von einer Partnertransaktion empfangen wird. Beim Empfang eines eingehenden Signals wird das Feld EIBSIG immer festgelegt.

Standardaktion: Die Bedingung wird ignoriert.

81 TERMERR

Tritt für einen sitzungsbezogenen Fehler auf. Jede Aktion außer einem Befehl FREE für diesen Dialog verursacht einen Abbruch mit dem Code ATCV.

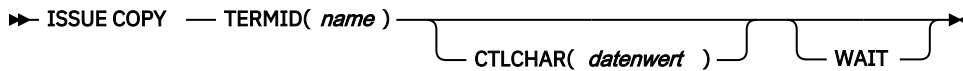
Eine Anforderung CANCEL TASK durch ein Benutzerknotenfehlerprogramm (NEP) kann diese Bedingung auslösen, wenn die Task eine ausstehende aktive Terminalsteueranforderung hat, wenn das Programm für abnormale Knotenbedingungen den Sitzungsfehler verarbeitet.

Standardaktion: Die Task wird abnormal mit dem Abbruchcode ATNI beendet.

ISSUE COPY (3270 logisch)

Kopiert Daten aus einer logischen 3270-Einheit.

ISSUE COPY (3270 logisch)



Bedingungen: LENGERR, NOTALLOC, TERMERR

Beschreibung

Der Befehl **ISSUE COPY** kopiert das Format und die Daten, die im Puffer eines angegebenen Terminals enthalten sind, in den Puffer des Terminals, das die Transaktion gestartet hat. Beide Terminals müssen an denselben DFV-Controller angeschlossen sein.

Optionen

CTLCHAR(datenwert)

Gibt ein 1 Byte großes Kopiersteuerzeichen (CCC - Copy Control Character) an, das die Kopierfunktion definiert. Ein COBOL-Benutzer muss einen Datenbereich angeben, der dieses Zeichen enthält. Wenn die Option nicht angegeben wird, wird der Inhalt des gesamten Puffers (einschließlich Nullen) kopiert.

TERMINID(name)

Gibt den Namen (1-4 Zeichen) des Terminals an, dessen Puffer kopiert werden soll. Das Terminal muss in der Terminalsteuertabelle (TCT) definiert worden sein.

WAIT

Gibt an, dass die Verarbeitung des Befehls abgeschlossen sein muss, bevor eine nachfolgende Verarbeitung versucht wird.

Wenn die Option **WAIT** nicht angegeben wird, wird die Steuerung an das Anwendungsprogramm zurückgegeben, sobald die Verarbeitung des Befehls gestartet wurde. Eine nachfolgende Ein- oder Ausgabeanforderung (Terminalsteuerung, BMS oder Batch Data Interchange) an das Terminal, das der Task zugeordnet ist, veranlasst das Anwendungsprogramm zu warten, bis die vorherige Anforderung abgeschlossen wurde.

Bedingungen

22 LENGERR

Tritt auf, wenn ein Wert außerhalb des gültigen Bereichs angegeben wird.

Standardaktion: Die Task wird abnormal beendet.

61 NOTALLOC

Tritt auf, wenn die im Befehl angegebene Funktion (Facility) nicht der Anwendung gehört.

Standardaktion: Die Task wird abnormal beendet.

81 TERMERR

Tritt für einen terminalbezogenen Fehler auf.

Eine Anforderung **CANCEL TASK** durch ein Benutzerknotenfehlerprogramm (NEP) kann diese Bedingung auslösen, wenn die Task eine ausstehende aktive Terminalsteueranforderung hat, wenn das Programm für abnormale Knotenbedingungen den Sitzungsfehler verarbeitet.

Standardaktion: Die Task wird abnormal mit dem Abbruchcode ATNI beendet.

ISSUE DISCONNECT (Standard)

Beendet eine Sitzung zwischen CICS und einer logischen Einheit oder einem Terminal.

ISSUE DISCONNECT (Standard)

➡ ISSUE DISCONNECT ➡

Bedingung: SIGNAL, TERMERR

Beschreibung

Der Befehl ISSUE DISCONNECT beendet Sitzungen zwischen CICS und den folgenden Terminals oder logischen Einheiten:

- Logische Anzeigeeinheit (LUTYPE2) - 3270
- Logische Druckereinheit (LUTYPE3) - 3270
- Logische Einheit vom LU-Typ 4 - LUTYPE4
- Logische SCS-Druckereinheit - 3270
- Datensichtgerät - 2260 oder 2265
- Logische Einheit - 3270
- Logische Pipeline-Einheit - 3600
- Logische Einheit - 3600(3601)
- Logische Einheit - 3600(3614)
- Werkskommunikationssystem - 3630
- Logische Interpreter-Einheit - 3650
- Logische Hostdialogeinheit (3270) - 3650
- Logische Hostdialogeinheit (3653) - 3650
- Logische Hostbefehlsprozessoreinheit - 3650(3680)
- Interaktive logische Einheit - 3767/3770
- Logische Einheit mit Stapelverarbeitung - 3770
- Logische Einheiten - 3790

Bedingungen

Für die meisten Typen von Terminals und logischen Einheiten löst der Befehl ISSUE DISCONNECT keine Bedingungen aus. Folgende Ausnahmen sind möglich:

24 SIGNAL

Tritt nur für einen Befehl ISSUE DISCONNECT für logische Einheiten der Typen LUTYPE4, 3600(3601), 3767 interaktiv, 3770 mit Stapelverarbeitung und 3790 mit vollem Funktionsumfang auf.

Diese Bedingung tritt auf, wenn ein eingehender Datenflusstuerbefehl SIGNAL von einer logischen Einheit oder Sitzung empfangen wird. Das Feld EIBSIG wird immer gesetzt, wenn ein eingehendes Signal empfangen wird.

Standardaktion: Die Bedingung wird ignoriert.

81 TERMERR

Tritt nur für einen Befehl ISSUE DISCONNECT für logische Einheiten des Typs LUTYPE4 auf.

Diese Bedingung tritt für einen terminalbezogenen Fehler auf, wie zum Beispiel einen Sitzungsfehler. Diese Bedingung gilt nur für SNA-verbundene logische Einheiten. Aufgrund der Asynchronizität dieser Bedingung sollte das Anwendungsprogramm mit dem Befehl SEND CONFIRM oder SYNCPOINT sicherstellen, dass alle ausstehenden Fehler behoben wurden, bevor es die Steuerung abgibt. Wenn Sie diese Bedingung behandeln wollen, müssen Sie zunächst einen Befehl FREE absetzen, um die Sitzung

freizugeben. Wenn Sie dies nicht tun, tritt eine Bedingung INVREQ auf und es kommt zu einer abnormalen Beendigung mit ATCV, wenn Sie diese Bedingung nicht behandeln.

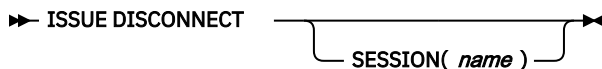
Eine Anforderung CANCEL TASK durch ein Benutzerknotenfehlerprogramm (NEP) kann diese Bedingung auslösen, wenn die Task eine ausstehende aktive Terminalsteueranforderung hat, wenn das Programm für abnormale Knotenbedingungen den Sitzungsfehler verarbeitet.

Standardaktion: Die Task wird abnormal mit dem Abbruchcode ATNI beendet.

ISSUE DISCONNECT (LUTYPE6.1)

Trennt eine logische Einheit vom LU-Typ 6.1 (LUTYPE6.1).

ISSUE DISCONNECT (LUTYPE6.1)



Bedingungen: NOTALLOC, TERMERR

Beschreibung

Der Befehl ISSUE DISCONNECT trennt die Verbindung zu der Einheit, wenn die Option DISCREQ=YES in der TYPETERM-Ressourcendefinition festgelegt ist.

Optionen

SESSION(name)

Gibt die symbolische ID (1 - 4 Zeichen) für einen Terminaleintrag in der Terminalsteuertabelle (TCTTE) für die Sitzung an. Diese Option gibt die alternative Funktion an, die getrennt werden soll. Wenn diese Option nicht angegeben wird, wird die Hauptfunktion für die Task getrennt.

Bedingungen

61 NOTALLOC

Tritt auf, wenn die im Befehl angegebene Funktion (Facility) nicht der Anwendung gehört.

Standardaktion: Die Task wird abnormal beendet.

81 TERMERR

Tritt für einen terminalbezogenen Fehler auf, wie zum Beispiel einen Sitzungsfehler.

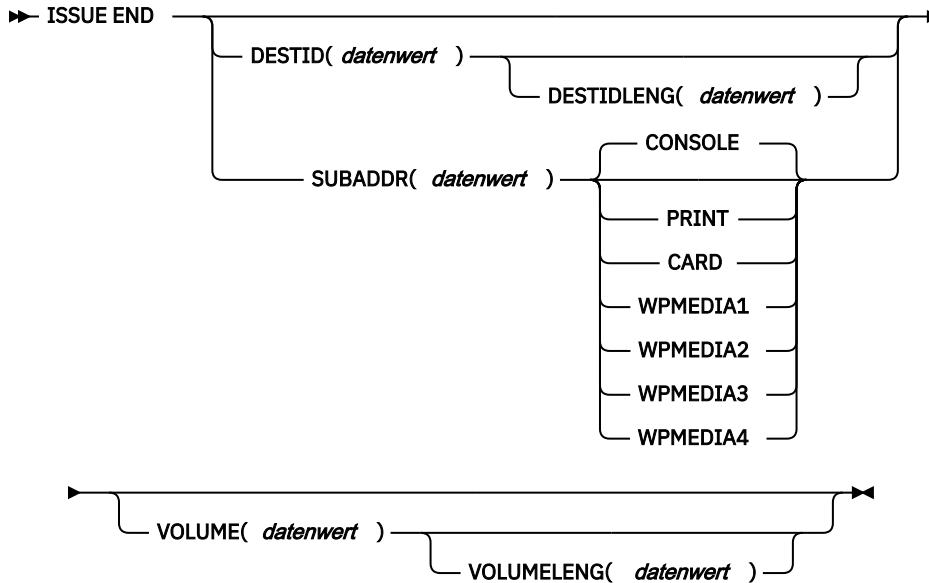
Diese Bedingung kann durch eine Anforderung CANCEL TASK durch ein Knotenfehlerprogramm des Benutzers verursacht werden, falls für die Task eine Terminalsteuerungsanforderung aussteht, wenn der Sitzungsfehler durch das Programm des Knotens für abnormale Bedingungen abgewickelt wird.

Standardaktion: Die Task wird abnormal mit dem Abbruchcode ATNI beendet.

ISSUE END

Beendet die Verarbeitung einer Datei (Dataset).

ISSUE END



Bedingungen: FUNCERR, INVREQ, SELNERR, UNEXPIN

Beschreibung

Der Befehl **ISSUE END** beendet die Kommunikation mit einer Datei (Dataset) in einem externen Controller oder in dem ausgewählten Medium. Die Datei, die in der Option **DESTID** angegeben wird, oder das ausgewählte Medium wird normal abgewählt. Die Optionen **CONSOLE**, **PRINT**, **CARD** und **WPMEDIA1–4** sind Alternativen für **DESTID** und **DESTIDLENG**.

Optionen

CARD

Gibt an, dass das Ausgabemedium ein Kartenlese- oder Kartenstanzgerät ist. Diese Option ist mit den Optionen **DESTID** und **DESTIDLENG** nicht gültig.

CONSOLE

Gibt an, dass das Ausgabemedium dasjenige ist, das für Nachrichten an den Bediener vorgesehen ist. Diese Option ist mit den Optionen **DESTID** und **DESTIDLENG** nicht gültig. Diese Option bezieht sich auf ein programmierbares Subsystem wie das Datenübertragungssystem IBM 3790. Sie bezieht sich nicht auf eine CICS- oder Systemkonsole.

DESTID(*datenwert*)

Gibt den Namen (1 - 8 Zeichen) der Datei in dem externen Ziel an.

DESTIDLENG(*datenwert*)

Gibt die Länge (Halbwort-Binärwert) des Namens an, der in der Option **DESTID** angegeben wird.

PRINT

Gibt an, dass das Ausgabemedium ein Drucker ist.

SUBADDR(*datenwert*)

Gibt die Unteradresse des Mediums als Halbwort-Binärwert (im Bereich von 0–15) an, die es ermöglicht, Medien desselben Typs wie zum Beispiel "printer 1" oder "printer 2" zu definieren. Der Wert 15 bedeutet ein Medium eines beliebigen Typs. Der Standardwert ist null.

VOLUME(*datenwert*)

Gibt den Namen (1 - 6 Zeichen) einer Diskette in einem externen Ziel an, die die Datei enthält, die in der Option **DESTID** angegeben wird.

VOLUMEENG(datenwert)

Gibt die Länge (Halbwort-Binärwert) des Namens an, der in der Option VOLUME angegeben wird.

WPMEDIA1 bis WPMEDIA4

Gibt für jede bestimmte Einheit vom LU-Typ 4 (LUTYPE4) an, dass ein Textverarbeitungsmedium in Bezug auf eine bestimmte Ein-/Ausgabeeinheit definiert ist.

Bedingungen**48 FUNCERR**

Wird ausgelöst, wenn ein Fehler bei der Ausführung des Befehls auftritt. Die Zielauswahl bleibt davon unbeeinträchtigt und andere Befehle für dasselbe Ziel werden möglicherweise erfolgreich ausgeführt.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

RESP2-Werte:

200

Eine Serveranwendung für Distributed Program Link hat in der Option CONVID die Sitzung für die Funktionsverlagerung (ihre Hauptfunktion) angegeben.

Standardaktion: Die Task wird abnormal beendet.

47 SELNERR

Wird ausgelöst, wenn ein Fehler bei der Zielauswahl auftritt. Das Ziel wird nicht ausgewählt und andere Befehle für dasselbe Ziel werden wahrscheinlich nicht erfolgreich ausgeführt.

Standardaktion: Die Task wird abnormal beendet.

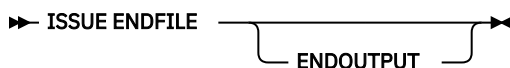
49 UNEXPIN

Tritt auf, wenn nicht erwartete oder nicht erkannte Informationen vom externen Controller empfangen werden.

Standardaktion: Die Task wird abnormal beendet.

ISSUE ENDFILE

Gibt die Dateiendebedingung für das Dateneingabesystem 3740 an.

ISSUE ENDFILE

Bedingungen: INVREQ, NOTALLOC

Beschreibung

Der Befehl ISSUE ENDFILE gibt die Dateiendebedingung für das 3740-System an.

Optionen**83 ENDOUTPUT**

Gibt die Ausgabeendebedingung und die Dateiendebedingung an.

Bedingungen

16 INVREQ

RESP2-Werte:

200

Eine Distributed Program Link-Serveranwendung hat versucht, in ihrer Funktionsverlagerungssitzung ihre Hauptfunktion zu senden.

Standardaktion: Die Task wird abnormal beendet.

61 NOTALLOC

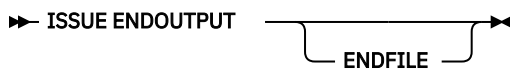
Tritt auf, falls die Anwendung nicht Eigner der im Befehl angegebenen Funktion ist.

Standardaktion: Die Task wird abnormal beendet.

ISSUE ENDOUTPUT

Gibt die Ausgabeendebedingung für das Dateneingabesystem 3740 an.

ISSUE ENDOUTPUT



Bedingungen: INVREQ, NOTALLOC

Beschreibung

Der Befehl ISSUE ENDOUTPUT gibt die Ausgabeendebedingung für das 3740-System an.

Optionen

20 ENDFILE

Gibt die Dateiendebedingung und die Ausgabeendebedingung an.

Bedingungen

16 INVREQ

RESP2-Werte:

200

Eine Distributed Program Link-Serveranwendung hat versucht, in ihrer Funktionsverlagerungssitzung ihre Hauptfunktion zu senden.

Standardaktion: Die Task wird abnormal beendet.

61 NOTALLOC

Tritt auf, falls die Anwendung nicht Eigner der im Befehl angegebenen Funktion ist.

Standardaktion: Die Task wird abnormal beendet.

ISSUE EODS

Sendet den Dateiende-Funktionsverwaltungsheader an die logische Einheit mit 3650-Interpreter.

ISSUE EODS



Bedingungen: INVREQ, NOTALLOC, TERMERR

Beschreibung

Der Befehl **ISSUE EODS** setzt einen Dateiende-Verwaltungsheader ab.

Bedingungen

16 INVREQ

RESP2-Werte:

200

Eine Distributed Program Link-Serveranwendung hat versucht, in ihrer Funktionsverlagerungssitzung ihre Hauptfunktion zu senden.

Standardaktion: Die Task wird abnormal beendet.

61 NOTALLOC

Tritt auf, falls die Anwendung nicht Eigner der im Befehl angegebenen Funktion ist.

Standardaktion: Die Task wird abnormal beendet.

81 TERMERR

Tritt für einen terminalbezogenen Fehler auf, wie zum Beispiel einen Sitzungsfehler.

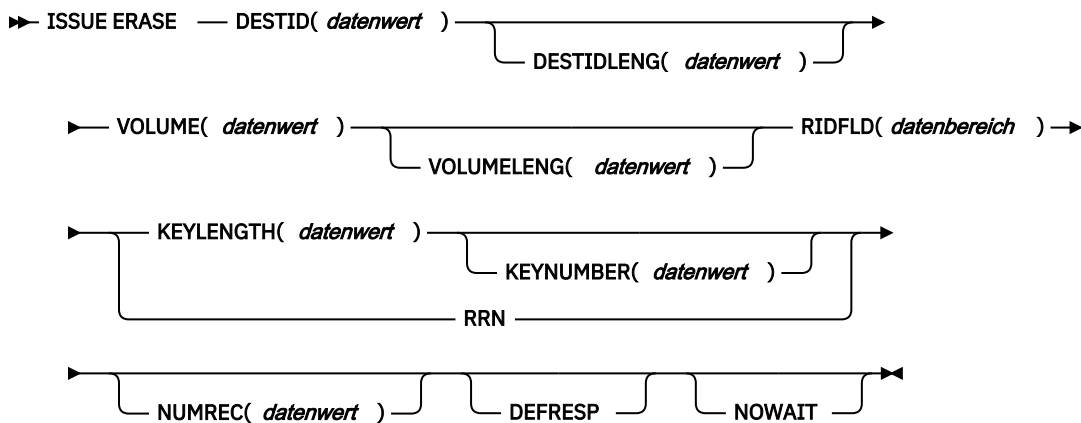
Eine Anforderung **CANCEL TASK** durch ein Benutzerknotenfehlerprogramm (NEP) kann diese Bedingung auslösen, wenn die Task eine ausstehende aktive Terminalsteueranforderung hat, wenn das Programm für abnormale Knotenbedingungen den Sitzungsfehler verarbeitet.

Standardaktion: Die Task wird abnormal mit dem Abbruchcode ATNI beendet.

ISSUE ERASE

Löscht einen Datensatz aus einer Datei (Dataset).

ISSUE ERASE



Bedingungen: FUNCERR, INVREQ, SELNERR, UNEXPIN

Beschreibung

Der Befehl **ISSUE ERASE** löscht einen Datensatz aus einer direkt organisierten Datei mit Schlüssel in einem externen Controller oder löscht einen Datensatz aus einer DPCX- oder DXAM-Datei für relative Sätze.

Optionen

DEFRESP

Gibt an, dass alle Terminalsteuerungsbefehle, die infolge des Befehls **ISSUE ERASE** abgesetzt werden, eine unbedingte Antwort vom externen Stapelverarbeitungsprogramm anfordern sollen, und dies unabhängig von der Angabe der Nachrichtenintegrität für die CICS-Task (durch den Systemprogrammierer).

DESTID(*datenwert*)

Gibt den Namen (1 - 8 Zeichen) der Datei in dem externen Ziel an.

DESTIDLENG(*datenwert*)

Gibt die Länge (Halbwort-Binärwert) des Namens an, der in der Option **DESTID** angegeben wird.

KEYLENGTH(*datenwert*)

Gibt die Länge des Schlüssels, der in der Option **RIDFLD** angegeben wird, als Halbwort-Binärwert an.

KEYNUMBER(*datenwert*)

Gibt die Nummer des Index als Halbwort-Binärwert an, der zum Aufsuchen des Datensatzes verwendet werden soll. Es können acht Indizes (1–8) angegeben werden. Der Standardwert ist 1. Diese Option gilt nur für **DPCX** oder **DXAM** und schließt sich mit **RRN** gegenseitig aus.

NOWAIT

Gibt an, dass die CICS-Task die Verarbeitung fortsetzt, ohne auf den Abschluss des Befehls **ISSUE ERASE** zu warten. Wenn diese Option nicht angegeben wird, wird die Taskaktivität bis zum Abschluss des Befehls ausgesetzt.

NUMREC(*datenwert*)

Gibt für eine Datei für relative Sätze (**RRD** - Relative Record Dataset) einen Halbwort-Binärwert für die Anzahl der logischen Datensätze an, die gelöscht werden sollen. Datensätze werden sequenziell ab dem Datensatz ersetzt, der durch die Option **RIDFLD** angegeben wird.

Für eine indexierte Datei kann **NUMREC** nicht angegeben werden, da nur ein Datensatz gelöscht wird.

RIDFLD(*datenbereich*)

Gibt das Feld für die Satzkennung an.

Für eine Datei für relative Sätze gibt die Option **RIDFLD** einen ganzzahligen Vollwort-Binärwert (die relative Satznummer des Datensatzes beginnend bei null) an, und die Option **RRN** wird verwendet.

Für eine indexierte Datei gibt die Option **RIDFLD** den Schlüssel an, der in die Daten eingebettet ist. Die Option **KEYLENGTH** ist ebenfalls erforderlich.

RRN

Gibt an, dass das Datensatz-ID-Feld, das in der Option **RIDFLD** angegeben wird, eine relative Satznummer (**RRN** - Relative Record Number) enthält. Wenn die Option nicht angegeben wird, wird angenommen, dass **RIDFLD** einen Schlüssel angibt.

VOLUME(*datenwert*)

Gibt den Namen (1 - 6 Zeichen) einer Diskette in einem externen Ziel an, die die Datei enthält, die in der Option **DESTID** angegeben wird.

VOLUMELENG(*datenwert*)

Gibt die Länge (Halbwort-Binärwert) des Namens an, der in der Option **VOLUME** angegeben wird.

Bedingungen

48 FUNCERR

Wird ausgelöst, wenn ein Fehler bei der Ausführung des Befehls auftritt. Die Zielauswahl bleibt davon unbeeinträchtigt und andere Befehle für dasselbe Ziel werden möglicherweise erfolgreich ausgeführt.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

RESP2-Werte:

200

Eine Serveranwendung für Distributed Program Link hat in der Option CONVID die Sitzung für die Funktionsverlagerung (ihre Hauptfunktion) angegeben.

Standardaktion: Die Task wird abnormal beendet.

47 SELNERR

Wird ausgelöst, wenn ein Fehler bei der Zielauswahl auftritt. Das Ziel wird nicht ausgewählt und andere Befehle für dasselbe Ziel werden wahrscheinlich nicht erfolgreich ausgeführt.

Standardaktion: Die Task wird abnormal beendet.

49 UNEXPIN

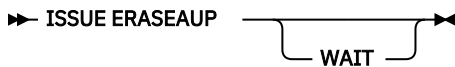
Tritt auf, wenn nicht erwartete oder nicht erkannte Informationen vom externen Controller empfangen werden.

Standardaktion: Die Task wird abnormal beendet.

ISSUE ERASEAUP

Löscht alle ungeschützten Felder aus einem 3270-Puffer.

ISSUE ERASEAUP



Bedingungen: INVREQ, NOTALLOC, TERMERR

Beschreibung

Der Befehl ISSUE ERASEAUP löscht ungeschützte Felder durch folgende Aktionen:

1. Inhalt aller ungeschützten Felder auf Nullen setzen (X'00')
2. Tags für modifizierte Daten in jedem ungeschützten Feld auf null zurücksetzen
3. Cursor in das erste ungeschützte Feld setzen
4. Tastatur wiederherstellen

Sie können den Befehl ISSUE ERASEAUP für die folgenden Typen von logischen Einheiten 3270 verwenden:

- Logische Anzeigeeinheit (LUTYPE2) - 3270
- Logische Druckereinheit (LUTYPE3) - 3270
- Logische Einheit - 3270
- Logische Hostdialogeinheit (3270) - 3650
- Logische Einheit - 3790 (3270-Anzeige)
- Logische Einheit - 3790 (3270-Drucker)

Optionen

WAIT

Stellt sicher, dass die Löschung abgeschlossen wird, bevor die Steuerung an das Anwendungsprogramm zurückgegeben wird. Wenn Sie WAIT nicht angeben, wird die Steuerung an das Anwendungsprogramm zurückgegeben, sobald die Verarbeitung des Befehls ISSUE ERASEAUP gestartet wird.

Bedingungen

16 INVREQ

RESP2-Werte:

200

Eine Serveranwendung für Distributed Program Link hat in der Option CONVID die Sitzung für die Funktionsverlagerung (ihre Hauptfunktion) angegeben.

Standardaktion: Die Task wird abnormal beendet.

61 NOTALLOC

Tritt auf, falls die Anwendung nicht Eigner der im Befehl angegebenen Funktion ist.

Standardaktion: Die Task wird abnormal beendet.

81 TERMERR

Wird ausgelöst, wenn ein terminalbezogener Fehler auftritt.

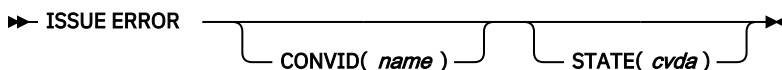
Eine Anforderung CANCEL TASK durch ein Benutzerknotenfehlerprogramm (NEP) kann diese Bedingung auslösen, wenn die Task eine ausstehende aktive Terminalsteueranforderung hat, wenn das Programm für abnormale Knotenbedingungen den Sitzungsfehler verarbeitet.

Standardaktion: Die Task wird abnormal mit dem Abbruchcode ATNI beendet.

ISSUE ERROR

Informiert den Partner des formatfreien APPC-Dialogs über einen Fehler.

ISSUE ERROR (APPC)



Bedingungen: INVREQ, NOTALLOC, SIGNAL, TERMERR

Beschreibung

Mit dem Befehl ISSUE ERROR kann ein Anwendungsprogramm einen Prozess in einem verbundenen APPC-System darüber informieren, dass ein vom Programm erkannter Fehler aufgetreten ist. Eine ferne CICS-Anwendung wird zum Beispiel dadurch benachrichtigt, dass das Feld EIBERR mit EIBERRCD=X'0889' festgelegt wird. Die Aktionen, die zur Wiederherstellung nach dem Fehler erforderlich sind, liegen in der Verantwortung der Logik, die in beiden Anwendungsprogrammen enthalten ist. Das Anwendungsprogramm kann diesen Befehl verwenden, um negativ zu antworten, wenn die Option CONFIRM in einem Befehl SEND angegeben wurde, der durch einen Prozess in einem verbundenen APPC-System ausgeführt wurde.

Optionen

CONVID(name)

Gibt den Dialog an, auf den sich der Befehl bezieht. Der vier Zeichen lange Name bezeichnet entweder das Token, das von einem zuvor ausgeführten Befehl ALLOCATE in EIBRSRCE im EIB zurückgegeben wurde, oder das Token, das die Hauptfunktion (die durch einen zuvor ausgeführten Befehl ASSIGN zurückgegeben wurde) darstellt.

Aus Gründen der Kompatibilität mit früheren Releases wird SESSION als Synonym für CONVID akzeptiert. Neue Programme sollten CONVID verwenden.

Wenn weder CONVID noch SESSION angegeben wird, wird die Hauptfunktion angenommen.

STATE(*cvda*)

Ruft den Status des aktuellen Datenaustausches ab. Die folgenden CVDA-Werte können von CICS zurückgegeben werden:

- ALLOCATED
- CONFFREE
- CONFRECEIVE
- CONFSEND
- FREE
- PENDFREE
- PENDRECEIVE
- RECEIVE
- ROLLBACK
- SEND
- SYNCFREE
- SYNCRECEIVE
- SYNCSEND

Bedingungen

16 INVREQ

RESP2-Werte:

200

Eine Distributed Program Link-Serveranwendung hat die Funktionsverlagerungssitzung in der Option CONVID angegeben.

Tritt außerdem in jedem der folgenden Fälle auf (RESP2 nicht festgelegt):

- Der Befehl ist für den verwendeten APPC-Dialogtyp nicht gültig.
- Der Befehl wurde für einen CPI Communications-Dialog abgesetzt.

Standardaktion: Die Task wird abnormal beendet.

61 NOTALLOC

Tritt auf, wenn sich der in CONVID angegebene Wert nicht auf einen Dialog bezieht, dessen Eigener die Anwendung ist.

Standardaktion: Die Task wird abnormal beendet.

24 SIGNAL

Tritt auf, wenn ein eingehender Datenflusssteuerbefehl SIGNAL von einer Partnertransaktion empfangen wird. Beim Empfang eines eingehenden Signals wird das Feld EIBSIG immer festgelegt.

Standardaktion: Die Bedingung wird ignoriert.

81 TERMERR

Tritt für einen sitzungsbezogenen Fehler auf. Jede Aktion außer einem Befehl FREE für diesen Dialog verursacht einen Abbruch mit dem Code ATCV.

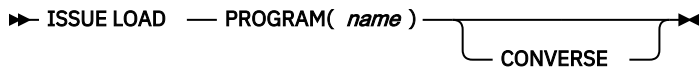
Eine Anforderung CANCEL TASK durch ein Benutzerknotenfehlerprogramm (NEP) kann diese Bedingung auslösen, wenn die Task eine ausstehende aktive Terminalsteueranforderung hat, wenn das Programm für abnormale Knotenbedingungen den Sitzungsfehler verarbeitet.

Standardaktion: Die Task wird abnormal mit dem Abbruchcode ATNI beendet.

ISSUE LOAD

Gibt auf einer logischen Einheit mit 3650-Interpreter den Namen eines Programms an.

ISSUE LOAD



Bedingungen: NONVAL, NOTALLOC, NOSTART, TERMERR

Beschreibung

Der Befehl ISSUE LOAD gibt den Namen des 3650-Anwendungsprogramms an, das geladen werden soll.

Optionen

CONVERSE

Gibt an, dass das 3650-Anwendungsprogramm mit dem Hostprozessor kommunizieren kann. Wenn diese Option nicht angegeben wird, kann das 3650-Anwendungsprogramm nicht mit dem Hostprozessor kommunizieren.

PROGRAM(name)

Gibt den Namen (1-8 Zeichen) des 3650-Anwendungsprogramms an, das geladen werden soll.

Bedingungen

09 NONVAL

Tritt auf, wenn der Name des 3650-Anwendungsprogramms nicht gültig ist.

Standardaktion: Die Task wird abnormal beendet.

10 NOSTART

Tritt auf, wenn das 3651-System das angeforderte 3650-Anwendungsprogramm nicht initialisieren kann.

Standardaktion: Die Task wird abnormal beendet.

61 NOTALLOC

Tritt auf, falls die Anwendung nicht Eigner der im Befehl angegebenen Funktion ist.

Standardaktion: Die Task wird abnormal beendet.

81 TERMERR

Tritt für einen terminalbezogenen Fehler auf.

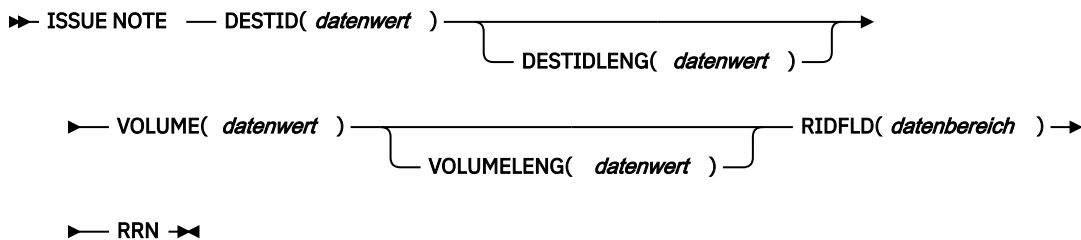
Eine Anforderung CANCEL TASK durch ein Benutzerknotenfehlerprogramm (NEP) kann diese Bedingung auslösen, wenn die Task eine ausstehende aktive Terminalsteueranforderung hat, wenn das Programm für abnormale Knotenbedingungen den Sitzungsfehler verarbeitet.

Standardaktion: Die Task wird abnormal mit dem Abbruchcode ATNI beendet.

ISSUE NOTE

Fordert die nächste Datensatznummer an.

ISSUE NOTE



Bedingungen: FUNCERR, INVREQ, SELNERR, UNEXPIN

Beschreibung

Der Befehl ISSUE NOTE fordert die Nummer des nächsten Datensatzes an. Er sucht die relative Satznummer des nächsten Datensatzes in einer direkt organisierte Datei mit Adressen. Die Nummer wird in dem Datenbereich zurückgegeben, der in der Option RIDFLD angegeben wird. Die Option RRN muss angegeben werden, weil mit einer relativen Satznummer gearbeitet wird.

Optionen

DESTID(*datenwert*)

Gibt den Namen (1 - 8 Zeichen) der Datei in dem externen Ziel an.

DESTIDLENG(*datenwert*)

Gibt die Länge (Halbwort-Binärwert) des Namens an, der in der Option DESTID angegeben wird.

RIDFLD(*datenbereich*)

Gibt einen Datenbereich als vier Zeichen langes Feld an, in dem die relative Satznummer des nächsten Datensatzes zurückgegeben wird.

RRN

Gibt an, dass das Datensatz-ID-Feld, das in der Option RIDFLD angegeben wird, eine relative Satznummer (RRN - Relative Record Number) enthält.

VOLUME(*datenwert*)

Gibt den Namen (1 - 6 Zeichen) einer Diskette in einem externen Ziel an, die die Datei enthält, die in der Option DESTID angegeben wird.

VOLUMELENG(*datenwert*)

Gibt die Länge (Halbwort-Binärwert) des Namens an, der in der Option VOLUME angegeben wird.

Bedingungen

48 FUNCERR

Wird ausgelöst, wenn ein Fehler bei der Ausführung des Befehls auftritt. Die Zielauswahl bleibt davon unbeeinträchtigt und andere Befehle für dasselbe Ziel werden möglicherweise erfolgreich ausgeführt.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

RESP2-Werte:

200

Eine Serveranwendung für Distributed Program Link hat in der Option CONVID die Sitzung für die Funktionsverlagerung (ihre Hauptfunktion) angegeben.

Standardaktion: Die Task wird abnormal beendet.

47 SELNERR

Wird ausgelöst, wenn ein Fehler bei der Zielauswahl auftritt. Das Ziel wird nicht ausgewählt und andere Befehle für dasselbe Ziel werden wahrscheinlich nicht erfolgreich ausgeführt.

Standardaktion: Die Task wird abnormal beendet.

49 UNEXPIN

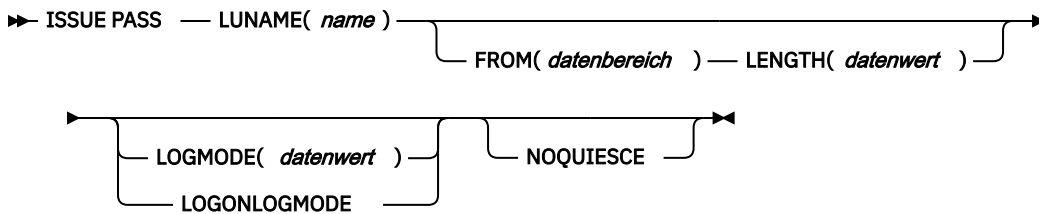
Tritt auf, wenn nicht erwartete oder nicht erkannte Informationen vom externen Controller empfangen werden.

Standardaktion: Die Task wird abnormal beendet.

ISSUE PASS

z/OS Communications Server-Anwendungsrouting.

ISSUE PASS



Bedingungen: INVREQ, LENGERR, NOTALLOC

Beschreibung

Der Befehl ISSUE PASS trennt das Terminal von CICS nach Beendigung der Task und überträgt es an die z/OS Communications Server-Anwendung, die in der Option LUNAME definiert wird.

Dieser Befehl setzt voraus, dass AUTH=PASS im z/OS Communications Server-APPL-Makro für das CICS-Terminaleignersystem codiert ist, das den Befehl absetzt, wobei DISCREQ=YES oder RELREQ=YES in der RDO-TYPETERM-Ressourcendefinition für ein Terminal angegeben ist, auf dem diese Funktion möglicherweise verwendet wird.

Wenn die Option LUNAME den Namen eines anderen CICS-Systems angibt, können Sie den Befehl EXTRACT LOGONMSG verwenden, um auf die Daten zuzugreifen, auf die durch diesen Befehl verwiesen wird.

Aufgrund einer Einschränkung in z/OS Communications Server ist die maximale Länge der Benutzerdaten auf 255 begrenzt.

Anmerkung: Durch den Systeminitialisierungsparameter CLSDSTP=NOTIFY|NONOTIFY können Sie das Knotenfehlerprogramm (NEP) und die Konsole benachrichtigen lassen, ob der PASS-Befehl erfolgreich war oder nicht. Das NEP kann so codiert werden, dass eine Sitzung, die durch einen nicht erfolgreichen PASS-Befehl beendet wurde, erneut hergestellt wird. Programmierinformationen dazu, wie dies erreicht werden kann, finden Sie im Abschnitt zum Knotenfehlerprogramm (NEP) unter [Knotenfehlerprogramm schreiben](#).

Optionen

FROM(datenbereich)

Gibt den Datenbereich an, der die Anmeldebenutzerdaten enthält, die an die Anwendung zu übergeben sind, die in der Option LUNAME genannt wird. Diese Option kann weggelassen werden, wenn die Option ATTACHID in einem LUTYPE6.1-Befehl angegeben wurde.

LENGTH(datenwert)

Gibt die Länge der ausgegebenen Daten als Halbwort-Binärwert an.

LOGMODE(datenwert)

Gibt den Namen (1-8 Zeichen) des z/OS Communications Server-Anmeldemodustabelleneintrags an, der von z/OS Communications Server zum Einrichten der neuen Sitzung verwendet wird.

LOGONLOGMODE

Gibt an, dass die neue Sitzung mit dem z/OS Communications Server-Anmeldemodustabelleneintrag eingerichtet werden soll, der verwendet wurde, als die Sitzung angemeldet wurde.

Anmerkung: Der gespeicherte Name in LOGMODE wird dem Steuervektor X'0D' in der z/OS Communications Server-CINIT-Anforderung entnommen. Dies ist der LOGMODE-Name, der in diesem System bekannt ist.

Wenn persistente Sitzungen (PSDINT=nnn in den SIT-Parametern) verwendet werden, sollte die TYPE-TERM-Definition für jedes Terminal, das durch ISSUE PASS übertragen werden soll, den Parameter RECOPTION(NONE) verwenden, da der LOGMODE-Anmeldename nicht über Neustarts persistenter Sitzungen hinweg wiederhergestellt wird.

Wenn weder LOGMODE noch LOGONLOGMODE angegeben wird, wird die neue Sitzung mit dem LOGMODE-Standardnamen eingerichtet.

LUNAME(name)

Gibt den Namen (1-8 Zeichen) der z/OS Communications Server-Anwendung an, an die das Terminal übergeben werden soll.

NOQUIESCE

Gibt an, dass der Benutzer eine Wiederherstellung nach bestimmten Übergabebefehlen durchführen kann.

Bedingungen

16 INVREQ

Tritt auf, wenn der Befehl für die verwendete logische Einheit nicht gültig ist.

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

Tritt auf, wenn ein Wert außerhalb des gültigen Bereichs in der Option LENGTH angegeben wird.

Standardaktion: Die Task wird abnormal beendet.

61 NOTALLOC

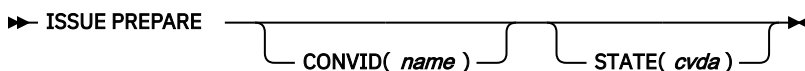
Tritt auf, falls die Anwendung nicht Eigner der im Befehl angegebenen Funktion ist.

Standardaktion: Die Task wird abnormal beendet.

ISSUE PREPARE

Setzt den ersten Ablauf einer Synchronisationspunktanforderung an einen formatfreien APPC-Dialog ab.

ISSUE PREPARE (APPC)



Bedingungen: INVREQ, NOTALLOC, TERMERR

Beschreibung

Der Befehl ISSUE PREPARE gilt nur für die verteilte Transaktionsverarbeitung über APPC-Verbindungen. Er gibt einem Synchronisationspunktinitiator die Möglichkeit, einen Synchronisationspunktteilnehmer für die Durchführung eines Synchronisationspunkts vorzubereiten, indem nur der erste Ablauf (Vorbereitung zur Festschreibung) des Synchronisationspunktaustausches gesendet wird. Abhängig von der Antwort des Synchronisationspunktteilnehmers, kann der Initiator mit dem Synchronisationspunkt fortfahren, indem er einen Befehl SYNCPOINT absetzt oder eine Zurücksetzung mit dem Befehl SYNCPOINT ROLL-BACK einleitet.

Optionen

CONVID(name)

Gibt den Dialog an, auf den sich der Befehl bezieht. Der vier Zeichen lange Name bezeichnet entweder das Token, das von einem zuvor ausgeführten Befehl ALLOCATE in EIBRSRCE im EIB zurückgegeben wurde, oder das Token, das die Hauptfunktion (die durch einen zuvor ausgeführten Befehl ASSIGN zurückgegeben wurde) darstellt.

Aus Gründen der Kompatibilität mit früheren Releases wird SESSION als Synonym für CONVID akzeptiert. Neue Programme sollten CONVID verwenden.

Wenn weder CONVID noch SESSION angegeben wird, wird die Hauptfunktion angenommen.

STATE(cvda)

Ruft den Status des aktuellen Datenaustausches ab. Die folgenden CVDA-Werte können von CICS zurückgegeben werden:

- ALLOCATED
- CONFFREE
- CONFRECEIVE
- CONFSEND
- FREE
- PENDFREE
- PENDRECEIVE
- RECEIVE
- ROLLBACK
- SEND
- SYNCFREE
- SYNCRECEIVE
- SYNCSEND

Bedingungen

16 INVREQ

RESP2-Werte:

200

Eine Serveranwendung für Distributed Program Link hat in der Option CONVID die Sitzung für die Funktionsverlagerung (ihre Hauptfunktion) angegeben.

Tritt außerdem in jedem der folgenden Fälle auf (RESP2 nicht festgelegt):

- Der Dialog ist kein formatfreier APPC-Dialog.
- Der Dialogstatus ist für die Anforderung nicht gültig.
- Die Synchronisationsebene des Dialogs ist nicht 2.

Standardaktion: Die Task wird abnormal beendet.

61 NOTALLOC

Tritt auf, wenn sich der Wert von CONVID im Befehl nicht auf einen Dialog bezieht, dessen Eigner die Anwendung ist.

Standardaktion: Die Task wird abnormal beendet.

81 TERMERR

Tritt für einen sitzungsbezogenen Fehler auf. Jede Aktion außer einem Befehl FREE für diesen Dialog verursacht einen Abbruch mit dem Code ATCV.

Eine Anforderung CANCEL TASK durch ein Benutzerknotenfehlerprogramm (NEP) kann diese Bedingung auslösen, wenn die Task eine ausstehende aktive Terminalsteueranforderung hat, wenn das Programm für abnormale Knotenbedingungen den Sitzungsfehler verarbeitet.

Standardaktion: Die Task wird abnormal mit dem Abbruchcode ATNI beendet.

ISSUE PRINT

Druckt die angezeigten Daten auf dem ersten verfügbaren Drucker aus.

ISSUE PRINT

➡ ISSUE PRINT ➡

Bedingungen: INVREQ, NOTALLOC, TERMERR

Beschreibung

Der Befehl ISSUE PRINT druckt angezeigte Daten auf dem ersten verfügbaren Drucker, der auf eine Druckeranforderung antworten kann.

Der Befehl ISSUE PRINT kann auf einer Reihe von logischen Einheiten mit den nachfolgend definierten Druckern verwendet werden:

- Für eine logische Einheit 3270 oder logische Einheit 3650 mit Hostdialog (3270) muss der Drucker durch die Optionen PRINTER oder ALTPRINTER in der RDO-TERMINAL-Ressourcendefinition oder durch einen Drucker, der durch das automatisch installierte Benutzerprogramm bereitgestellt wird, definiert werden.
- Für eine logische Einheit mit 3270-Anzeige mit der Funktion PTRADAPT, die mit einer Einheit 3274 oder 3276 verwendet wird, wird der Drucker durch die Druckerberechtigungsmatrix zugeordnet. Die Funktion PTRADAPT wird durch die Angaben DEVICE=LUTYPE2 und PRINTADAPTER=YES in der RDO-TYPE-TERM-Ressourcendefinition aktiviert.
- Für eine logische Einheit 3790 (3270-Anzeige) wird der Drucker durch die Einheit 3790 zugeordnet.

Der Drucker muss in Betrieb sein, darf nicht momentan einer Task zugeordnet sein und muss zu demselben CICS-System gehören, zu dem auch das Terminal gehört, das die Transaktion ausführt.

Bedingungen

16 INVREQ

RESP2-Werte:

200

Eine Serveranwendung für Distributed Program Link hat in der Option CONVID die Sitzung für die Funktionsverlagerung (ihre Hauptfunktion) angegeben.

Standardaktion: Die Task wird abnormal beendet.

61 NOTALLOC

Tritt auf, falls die Anwendung nicht Eigner der im Befehl angegebenen Funktion ist.

Standardaktion: Die Task wird abnormal beendet.

81 TERMERR

Tritt für einen terminalbezogenen Fehler auf.

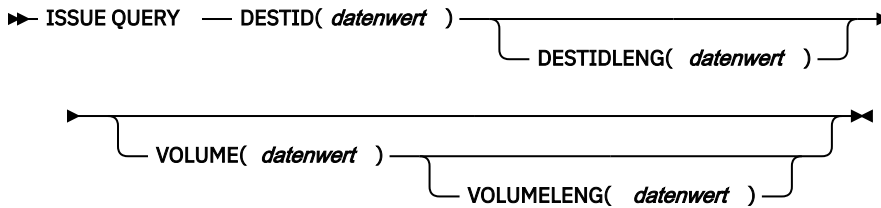
Eine Anforderung CANCEL TASK durch ein Benutzerknotenfehlerprogramm (NEP) kann diese Bedingung auslösen, wenn die Task eine ausstehende aktive Terminalsteueranforderung hat, wenn das Programm für abnormale Knotenbedingungen den Sitzungsfehler verarbeitet.

Standardaktion: Die Task wird abnormal mit dem Abbruchcode ATNI beendet.

ISSUE QUERY

Fragt eine Datei (Dataset) ab.

ISSUE QUERY



Bedingungen: FUNCERR, INVREQ, SELNERR, UNEXPIN

Beschreibung

Der Befehl ISSUE QUERY fragt eine Datei (Dataset) ab. Er dient zur Anforderung, dass eine sequenzielle Datei in einem externen Controller an das Hostsystem übertragen wird. Das Anwendungsprogramm sollte nach diesem Befehl die resultierenden eingehenden Daten mit Befehlen ISSUE RECEIVE abrufen oder die Transaktion beenden, damit CICS eine neue Transaktion zur Verarbeitung der Daten starten kann.

Optionen

DESTID(*datenwert*)

Gibt den Namen (1 - 8 Zeichen) der Datei in dem externen Ziel an.

DESTIDLENG(*datenwert*)

Gibt die Länge (Halbwort-Binärwert) des Namens an, der in der Option DESTID angegeben wird.

VOLUME(*datenwert*)

Gibt den Namen (1 - 6 Zeichen) einer Diskette in einem externen Ziel an, die die Datei enthält, die in der Option DESTID angegeben wird.

VOLUMELENG(*datenwert*)

Gibt die Länge (Halbwort-Binärwert) des Namens an, der in der Option VOLUME angegeben wird.

Bedingungen

48 FUNCERR

Wird ausgelöst, wenn ein Fehler bei der Ausführung des Befehls auftritt. Die Zielauswahl bleibt davon unbeeinträchtigt und andere Befehle für dasselbe Ziel werden möglicherweise erfolgreich ausgeführt.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

RESP2-Werte:

200

Eine Serveranwendung für Distributed Program Link hat in der Option CONVID die Sitzung für die Funktionsverlagerung (ihre Hauptfunktion) angegeben.

Standardaktion: Die Task wird abnormal beendet.

47 SELNERR

Wird ausgelöst, wenn ein Fehler bei der Zielauswahl auftritt. Das Ziel wird nicht ausgewählt und andere Befehle für dasselbe Ziel werden wahrscheinlich nicht erfolgreich ausgeführt.

Standardaktion: Die Task wird abnormal beendet.

49 UNEXPIN

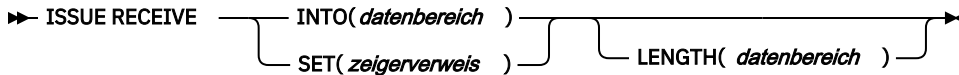
Tritt auf, wenn nicht erwartete oder nicht erkannte Informationen vom externen Controller empfangen werden.

Standardaktion: Die Task wird abnormal beendet.

ISSUE RECEIVE

Liest einen Datensatz aus einer Datei (Dataset).

ISSUE RECEIVE



Bedingungen: DSSTAT, EOC, EODS, INVREQ, LENGERR, UNEXPIN

Beschreibung

Der Befehl ISSUE RECEIVE liest eine sequenzielle Datei in einem externen Controller.

Die Option INTO gibt den Bereich an, in den die Daten eingelesen werden sollen. Die Option LENGTH muss einen Datenbereich angeben, der die maximale Datensatzlänge enthält, die das Programm akzeptiert. Wenn die Datensatzlänge die angegebene Maximallänge überschreitet, wird der Datensatz abgeschnitten und die Bedingung LENGERR tritt auf. Nach der Abrufoperation wird der in der Option LENGTH angegebene Datenbereich auf die Datensatzlänge (vor dem Abschneiden) gesetzt.

Alternativ kann ein Zeigerverweis in der Option SET angegeben werden. CICS fordert dann einen Bereich ausreichender Größe zur Aufnahme des Datensatzes an und setzt den Zeigerverweis auf die Adresse dieses Bereichs. Nach der Abrufoperation wird der Datenbereich, der in der Option LENGTH angegeben ist, auf die Datensatzlänge gesetzt.

Der externe Controller sendet möglicherweise die Daten nicht aus der Datei, die in dem Befehl ISSUE QUERY angegeben ist. Der Befehl ASSIGN muss verwendet werden, um den Wert von DESTID (der die Datei identifiziert, die übertragen wurde) und den Wert von DESTIDLENG (der die Länge der ID in DESTID angibt) abzurufen.

Optionen

INTO(datenbereich)

Gibt das empfangende Feld für die Daten an, die aus der Datei gelesen werden.

Wenn Sie den Befehl ISSUE RECEIVE mit der Option INTO angeben, muss der Parameter auf einen Datenbereich verweisen, der die maximale Länge der Daten angibt, zur deren Verarbeitung das Programm vorbereitet wurde. Wenn der angegebene Wert kleiner null ist, wird null angenommen. Wenn die Länge der Daten den angegebenen Wert überschreitet, werden die Daten auf diesen Wert abgeschnitten und die Bedingung LENGERR tritt auf. Nach Abschluss der Abrufoperation wird der Datenbereich auf die ursprüngliche Länge der Daten gesetzt.

LENGTH(datenbereich)

Gibt die Länge (Halbwort-Binärwert) der empfangenen Daten an.

Wenn Sie die Option SET angeben, müssen Sie auch die Option LENGTH angeben.

SET(zeigerverweis)

Gibt den Zeigerverweis an, der auf die Adressposition der Daten gesetzt werden soll, die aus der Datei gelesen werden.

Wenn Sie die Option SET angeben, muss der Parameter auf einen Datenbereich verweisen. Nach Abschluss der Abrufoperation wird der Datenbereich auf die Länge der Daten gesetzt.

Wenn dem Anwendungsprogramm DATALOCATION(ANY) zugeordnet ist, kann die Adresse der Daten ober- oder unterhalb der 16-MB-Grenze liegen.

Wenn dem Anwendungsprogramm DATALOCATION(BELOW) zugeordnet ist und sich die Daten oberhalb der 16-MB-Grenze befinden, werden die Daten an eine Adresse unterhalb der 16-MB-Grenze kopiert und die Adresse dieser Kopie wird zurückgegeben.

Wenn für die aktive Task TASKDATAKEY(USER) angegeben wird und der Speicherschutz aktiv ist, befinden sich die zurückgegebenen Daten in einem Benutzerschlüssel. Wenn für die aktive Task TASKDATAKEY(CICS) angegeben wird und der Speicherschutz aktiv ist, befinden sich die zurückgegebenen Daten in einem CICS-Schlüssel.

Wenn Sie die Option SET angeben, müssen Sie auch die Option LENGTH angeben.

Bedingungen

46 DSSTAT

Tritt auf, wenn sich der Zielstatus auf eine der folgenden Arten ändert:

- Der Datenstrom wird abnormal beendet.
- Der Datenstrom wird ausgesetzt.

Standardaktion: Die Task wird abnormal beendet.

06 EOC

Tritt auf, wenn die Anforderung/Antwort-Einheit (RU) mit gesetztem Anzeiger für Kettenende (EOC - End-of-Chain) empfangen wird. Das Feld EIBEOC enthält diesen Anzeiger ebenfalls.

Standardaktion: Die Bedingung wird ignoriert.

05 EODS

Tritt auf, wenn das Ende der Datei angetroffen wird.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

RESP2-Werte:

200

Eine Serveranwendung für Distributed Program Link hat in der Option CONVID die Sitzung für die Funktionsverlagerung (ihre Hauptfunktion) angegeben.

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

Tritt auf, wenn die Länge der abgerufenen Daten größer als der Wert ist, der durch die Option LENGTH angegeben wurde.

Standardaktion: Die Task wird abnormal beendet.

49 UNEXPIN

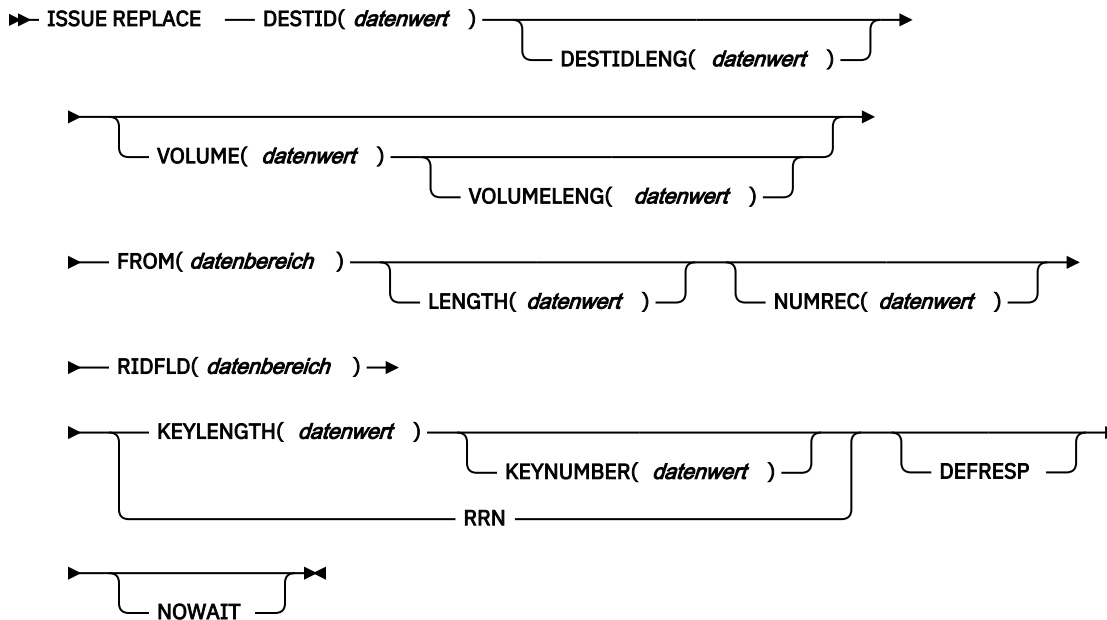
Tritt auf, wenn nicht erwartete oder nicht erkannte Informationen vom externen Controller empfangen werden.

Standardaktion: Die Task wird abnormal beendet.

ISSUE REPLACE

Aktualisiert einen Datensatz in einer Datei (Dataset).

ISSUE REPLACE



Bedingungen: FUNCERR, INVREQ, SELNERR, UNEXPIN

Beschreibung

Der Befehl **ISSUE REPLACE** aktualisiert (ersetzt) einen Datensatz in einer direkt organisierten Datei mit Adressen oder mit Schlüssel in einem externen Controller.

Optionen

DEFRESP

Gibt an, dass alle Terminalsteuerungsbefehle, die infolge des Befehls **ISSUE REPLACE** abgesetzt werden, eine unbedingte Antwort vom externen Stapelverarbeitungsprogramm anfordern, und dies unabhängig von der Angabe der Nachrichtenintegrität für die CICS-Task (durch den Systemprogrammierer).

DESTID(datenwert)

Gibt den Namen (1 - 8 Zeichen) der Datei in dem externen Ziel an.

DESTIDLENG(datenwert)

Gibt die Länge (Halbwort-Binärwert) des Namens an, der in der Option **DESTID** angegeben wird.

FROM(datenbereich)

Gibt die Daten an, die in die Datei zu schreiben sind.

KEYLENGTH(datenwert)

Gibt die Länge (Halbwort-Binärwert) des Schlüssels an, der in der Option **RIDFLD** angegeben wird.

KEYNUMBER(datenwert)

Gibt die Nummer des Index als Halbwort-Binärwert an, der zum Aufsuchen des Datensatzes verwendet werden soll. Es können acht Indizes (1 - 8) angegeben werden. Der Standardwert ist 1. Diese Option gilt nur für **DPCX/DXAM** und schließt sich mit **RRN** gegenseitig aus.

LENGTH(datenwert)

Gibt die Länge (Halbwort-Binärwert) der zu schreibenden Daten an.

NOWAIT

Gibt an, dass die CICS-Task die Verarbeitung fortsetzt, ohne auf den Abschluss des Befehls **ISSUE REPLACE** zu warten. Wenn diese Option nicht angegeben wird, wird die Taskaktivität bis zum Abschluss des Befehls ausgesetzt.

NUMREC(datenwert)

Gibt für eine Datei für relative Sätze (RRD - Relative Record Dataset) einen Halbwort-Binärwert für die Anzahl der logischen Datensätze an, die ersetzt werden sollen. Datensätze werden sequenziell ab dem Datensatz ersetzt, der durch die Option **RIDFLD** angegeben wird.

Für eine indexierte Datei kann **NUMREC** nicht angegeben werden, da nur ein Datensatz ersetzt wird.

RIDFLD(datenbereich)

Gibt das Feld für die Satzkennung an.

Für eine Datei für relative Sätze gibt die Option **RIDFLD** einen ganzzahligen Vollwort-Binärwert (die relative Satznummer des Datensatzes beginnend bei null) an, und die Option **RRN** wird verwendet.

Für eine indexierte Datei gibt die Option **RIDFLD** den Schlüssel an, der in die Daten eingebettet ist, die durch die Option **FROM** angegeben werden. Die Option **KEYLENGTH** ist ebenfalls erforderlich.

RRN

Gibt an, dass das Datensatz-ID-Feld, das in der Option **RIDFLD** angegeben wird, eine relative Satznummer (**RRN** - Relative Record Number) enthält. Diese Option ist für eine Datei für relative Sätze erforderlich.

Wenn die Option nicht angegeben wird, wird angenommen, dass **RIDFLD** einen Schlüssel angibt.

VOLUME(datenwert)

Gibt den Namen (1 - 6 Zeichen) einer Diskette in einem externen Ziel an, die die Datei enthält, die in der Option **DESTID** angegeben wird.

VOLUMELENG(datenwert)

Gibt die Länge (Halbwort-Binärwert) des Namens an, der in der Option **VOLUME** angegeben wird.

Bedingungen

48 FUNCERR

Wird ausgelöst, wenn ein Fehler bei der Ausführung des Befehls auftritt. Die Zielauswahl bleibt davon unbeeinträchtigt und andere Befehle für dasselbe Ziel werden möglicherweise erfolgreich ausgeführt.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

RESP2-Werte:

200

Eine Serveranwendung für Distributed Program Link hat in der Option **CONVID** die Sitzung für die Funktionsverlagerung (ihre Hauptfunktion) angegeben.

Standardaktion: Die Task wird abnormal beendet.

47 SELNERR

Wird ausgelöst, wenn ein Fehler bei der Zielauswahl auftritt. Das Ziel wird nicht ausgewählt und andere Befehle für dasselbe Ziel werden wahrscheinlich nicht erfolgreich ausgeführt.

Standardaktion: Die Task wird abnormal beendet.

49 UNEXPIN

Tritt auf, wenn nicht erwartete oder nicht erkannte Informationen vom externen Controller empfangen werden.

Standardaktion: Die Task wird abnormal beendet.

ISSUE RESET

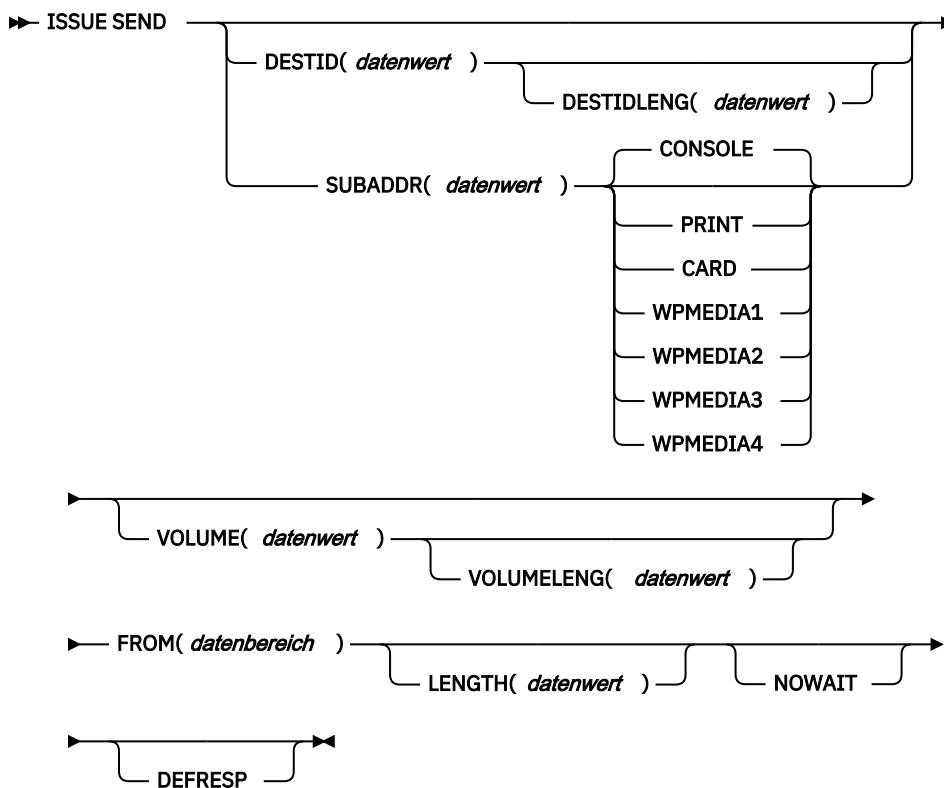
Gibt die Verwendung einer Telekommunikationsleitung frei.

Dieser Befehl wird aus Gründen der Kompatibilität mit früheren Releases von CICS unterstützt. Er wurde durch den Befehl [ISSUE DISCONNECT](#) ersetzt, dessen Verwendung empfohlen wird.

ISSUE SEND

Sendet Daten an eine benannte Datei (Dataset) oder an ein ausgewähltes Medium.

ISSUE SEND



Bedingungen: FUNCERR, IGREQCD, INVREQ, SELNERR, UNEXPIN

Beschreibung

Der Befehl **ISSUE SEND** sendet Daten an eine benannte Datei in einem externen Controller oder an ein ausgewähltes Medium in einer logischen Einheit mit Stapelverarbeitung oder eine logische Einheit vom LU-Typ 4 (LUTYPE4). Die Optionen **CONSOLE**, **PRINT**, **CARD** und **WPMEDIA1–4** sind Alternativen für **DESTID** und **DESTIDLENG**.

Optionen

CARD

Gibt an, dass das Ausgabemedium ein Kartenlese- oder Kartenstanzgerät ist. Diese Option ist mit den Optionen **DESTID** und **DESTIDLENG** nicht gültig.

CONSOLE

Gibt an, dass das Ausgabemedium dasjenige ist, das für Nachrichten an den Bediener vorgesehen ist. Diese Option ist mit den Optionen **DESTID** und **DESTIDLENG** nicht gültig. Diese Option bezieht sich auf

ein programmierbares Subsystem wie das Datenübertragungssystem IBM 3790. Sie bezieht sich nicht auf eine CICS- oder Systemkonsole.

DEFRESP

Gibt an, dass alle Terminalsteuerungsbefehle, die infolge des Befehls **ISSUE SEND** abgesetzt werden, eine unbedingte Antwort vom externen Stapelverarbeitungsprogramm anfordern, und dies unabhängig von der Angabe der Nachrichtenintegrität für die CICS-Task (durch den Systemprogrammierer).

DESTID(datenwert)

Gibt den Namen (1 - 8 Zeichen) der Datei in dem externen Ziel an.

DESTIDLENG(datenwert)

Gibt die Länge (Halbwort-Binärwert) des Namens an, der in der Option **DESTID** angegeben wird.

FROM(datenbereich)

Gibt die Daten an, die in die Datei zu schreiben sind.

LENGTH(datenwert)

Gibt einen Halbwort-Binärwert für die Länge der zu schreibenden Daten an.

NOWAIT

Gibt an, dass die CICS-Task die Verarbeitung fortsetzt, ohne auf den Abschluss des Befehls **ISSUE SEND** zu warten. Wenn diese Option nicht angegeben wird, wird die Taskaktivität bis zum Abschluss des Befehls ausgesetzt.

PRINT

Gibt an, dass die Ausgabe an ein Druckermedium erfolgt.

SUBADDR(datenwert)

Gibt die Unteradresse des Mediums als Halbwort-Binärwert (im Bereich von 0–15) an, die es ermöglicht, Medien desselben Typs wie zum Beispiel "printer 1" oder "printer 2" zu definieren. Der Wert 15 bedeutet ein Medium eines beliebigen Typs. Der Standardwert ist null.

VOLUME(datenwert)

Gibt den Namen (1 - 6 Zeichen) einer Diskette in einem externen Ziel an, die die Datei enthält, die in der Option **DESTID** angegeben wird.

VOLUMELENG(datenwert)

Gibt die Länge des Namens, der in der Option **VOLUME** angegeben wird, als Halbwort-Binärwert an.

WPMEDIA1 bis WPMEDIA4

Gibt für jede bestimmte Einheit vom LU-Typ 4 (**LUTYPE4**) an, dass ein Textverarbeitungsmedium in Bezug auf eine bestimmte Ein-/Ausgabeeinheit definiert ist.

Bedingungen

48 FUNCERR

Wird ausgelöst, wenn ein Fehler bei der Ausführung des Befehls auftritt. Die Zielauswahl bleibt davon unbeeinträchtigt und andere Befehle für dasselbe Ziel werden möglicherweise erfolgreich ausgeführt.

Standardaktion: Die Task wird abnormal beendet.

57 IGREQCD

Tritt auf, wenn ein Versuch unternommen wird, einen Befehl **ISSUE SEND** auszuführen, nachdem ein Datenflusssteuerbefehl **SIGNAL** mit einem Richtungswechselanforderungscode (**RCD** - Request Change Direction) von einer logischen Einheit mit LU-Typ 4 (**LUTYPE4**) empfangen wurde.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

RESP2-Werte:

200

Eine Serveranwendung für Distributed Program Link hat in der Option **CONVID** die Sitzung für die Funktionsverlagerung (ihre Hauptfunktion) angegeben.

Standardaktion: Die Task wird abnormal beendet.

47 SELNERR

Wird ausgelöst, wenn ein Fehler bei der Zielauswahl auftritt. Das Ziel wird nicht ausgewählt und andere Befehle für dasselbe Ziel werden wahrscheinlich nicht erfolgreich ausgeführt.

Standardaktion: Die Task wird abnormal beendet.

49 UNEXPIN

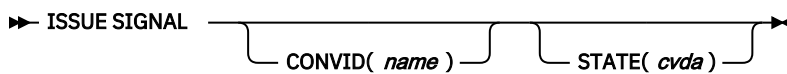
Tritt auf, wenn nicht erwartete oder nicht erkannte Informationen vom externen Controller empfangen werden.

Standardaktion: Die Task wird abnormal beendet.

ISSUE SIGNAL (APPC)

Fordert einen Richtungswechsel von der sendenden Transaktion in einem formatfreien APPC-Dialog an.

ISSUE SIGNAL (APPC)



Bedingungen: INVREQ, NOTALLOC, TERMERR

Beschreibung

Der Befehl ISSUE SIGNAL signalisiert in einem Transaktionsempfangsmodus der sendenden Transaktion, dass ein Moduswechsel erforderlich ist. Er löst die Bedingung SIGNAL beim nächsten Befehl SEND, RECEIVE oder CONVERSE aus, der in der sendenden Transaktion ausgeführt wird, und ein zuvor ausgeführter Befehl HANDLE CONDITION kann dazu verwendet werden, entweder eine Aktion auszuführen oder die Anforderung zu ignorieren.

Optionen

CONVID(name)

Gibt den Dialog an, auf den sich der Befehl bezieht. Der vier Zeichen lange Name bezeichnet entweder das Token, das von einem zuvor ausgeführten Befehl ALLOCATE in EIBRSRCE im EIB zurückgegeben wurde, oder das Token, das die Hauptfunktion (die durch einen zuvor ausgeführten Befehl ASSIGN zurückgegeben wurde) darstellt.

Aus Gründen der Kompatibilität mit früheren Releases wird SESSION als Synonym für CONVID akzeptiert. Neue Programme sollten CONVID verwenden.

Wenn weder CONVID noch SESSION angegeben wird, wird die Hauptfunktion angenommen.

STATE(cvda)

Ruft den Status des aktuellen Datenaustausches ab. Die folgenden CVDA-Werte können von CICS zurückgegeben werden:

- ALLOCATED
- CONFFREE
- CONFRECEIVE
- CONFSEND
- FREE
- PENDFREE
- PENDRECEIVE
- RECEIVE

- ROLLBACK
- SEND
- SYNCFREE
- SYNCRECEIVE
- SYNCSEND

Bedingungen

16 INVREQ

RESP2-Werte:

200

Eine Serveranwendung für Distributed Program Link hat in der Option CONVID die Sitzung für die Funktionsverlagerung (ihre Hauptfunktion) angegeben.

Tritt außerdem in jedem der folgenden Fälle auf (RESP2 nicht festgelegt):

- Der Befehl wurde in einem APPC-Dialog verwendet, der nicht die EXEC CICS-Schnittstelle verwendet und kein formatfreier Dialog ist.

Standardaktion: Die Task wird abnormal beendet.

61 NOTALLOC

Tritt auf, wenn sich der in CONVID angegebene Wert nicht auf einen Dialog bezieht, dessen Eigner die Anwendung ist.

Standardaktion: Die Task wird abnormal beendet.

81 TERMERR

Tritt für einen sitzungsbezogenen Fehler auf. Jede Aktion außer einem Befehl FREE für diesen Dialog verursacht einen Abbruch mit dem Code ATCV.

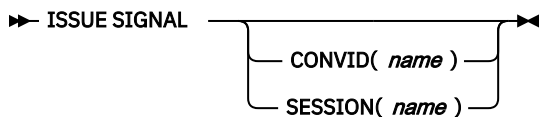
Eine Anforderung CANCEL TASK durch ein Benutzerknotenfehlerprogramm (NEP) kann die Bedingung TERMERR auslösen, wenn die Task eine ausstehende Terminalsteueranforderung hat, wenn das Programm für abnormale Knotenbedingungen den Sitzungsfehler verarbeitet.

Standardaktion: Die Task wird abnormal mit dem Abbruchcode ATNI beendet.

ISSUE SIGNAL (LUTYPE6.1)

Fordert einen Richtungswechsel von der sendenden Transaktion in einem LUTYPE6.1-Dialog an.

ISSUE SIGNAL (LUTYPE6.1)



Bedingungen: NOTALLOC, TERMERR

Beschreibung

Der Befehl ISSUE SIGNAL signalisiert in einem Transaktionsempfangsmodus der sendenden Transaktion, dass ein Moduswechsel erforderlich ist. Er löst die Bedingung SIGNAL beim nächsten Befehl SEND, RECEIVE oder CONVERSE aus, der in der sendenden Transaktion ausgeführt wird, und ein zuvor ausgeführter Befehl HANDLE CONDITION kann dazu verwendet werden, entweder eine Aktion auszuführen oder die Anforderung zu ignorieren.

Wenn weder CONVID noch SESSION angegeben wird, wird die Hauptfunktion für die Task verwendet.

Optionen

CONVID(*name*)

Gibt den Dialog an, auf den sich der Befehl bezieht. Der vier Zeichen lange Name bezeichnet entweder das Token, das von einem zuvor ausgeführten Befehl ALLOCATE in EIBRSRCE im EIB zurückgegeben wurde, oder das Token, das die Hauptfunktion (die durch einen zuvor ausgeführten Befehl ASSIGN zurückgegeben wurde) darstellt.

SESSION(*name*)

Gibt die symbolische ID (1 - 4 Zeichen) für einen Terminaleintrag in der Terminalsteuertabelle (TCTTE) für die Sitzung an. Diese Option gibt die alternative Funktion an, die verwendet werden soll.

Bedingungen

61 NOTALLOC

Tritt auf, falls die Anwendung nicht Eigner der im Befehl angegebenen Funktion ist.

Standardaktion: Die Task wird abnormal beendet.

81 TERMERR

Tritt für einen sitzungsbezogenen Fehler auf. Jede Aktion außer einem Befehl FREE für diesen Dialog verursacht einen Abbruch mit dem Code ATCV.

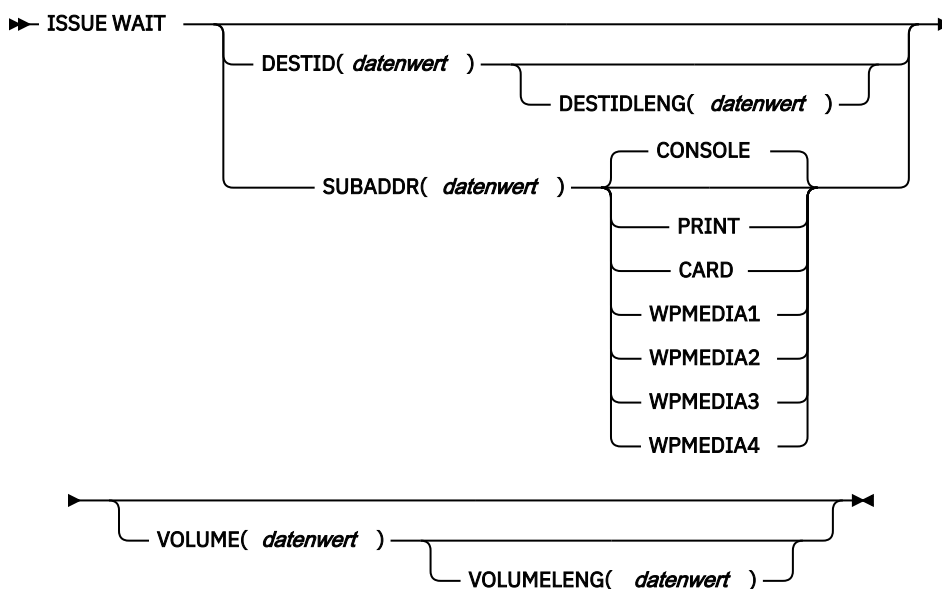
Eine Anforderung CANCEL TASK durch ein Benutzerknotenfehlerprogramm (NEP) kann die Bedingung TERMERR auslösen, wenn die Task eine ausstehende Terminalsteueranforderung hat, wenn das Programm für abnormale Knotenbedingungen den Sitzungsfehler verarbeitet.

Standardaktion: Die Task wird abnormal mit dem Abbruchcode ATNI beendet.

ISSUE WAIT

Wartet auf den Abschluss einer Operation.

ISSUE WAIT



Bedingungen: FUNCERR, INVREQ, SELNERR, UNEXPIN

Beschreibung

Der Befehl **ISSUE WAIT** setzt die Taskaktivität aus, bis der vorherige Batch Data Interchange-Befehl abgeschlossen ist. Dieser Befehl hat nur dann eine Bedeutung, wenn er in Anschluss an einen Befehl **ISSUE ADD**, **ISSUE ERASE**, **ISSUE REPLACE** oder **ISSUE SEND** abgesetzt wird. Die Optionen **CONSOLE**, **PRINT**, **CARD** und **WPMEDIA1–4** sind Alternativen für **DESTID** und **DESTIDLENG**.

Optionen

CARD

Gibt an, dass das Ausgabemedium ein Kartenlese- oder Kartenstanzgerät ist. Diese Option ist mit den Optionen **DESTID** und **DESTIDLENG** nicht gültig.

CONSOLE

Gibt an, dass das Ausgabemedium dasjenige ist, das für Nachrichten an den Bediener vorgesehen ist. Diese Option ist mit den Optionen **DESTID** und **DESTIDLENG** nicht gültig.

Diese Option bezieht sich auf ein programmierbares Subsystem wie das Datenübertragungssystem IBM 3790. Sie bezieht sich nicht auf eine CICS- oder Systemkonsole.

DESTID(datenwert)

Gibt den Namen (1 - 8 Zeichen) der Datei in dem externen Ziel an.

DESTIDLENG(datenwert)

Gibt die Länge (Halbwort-Binärwert) des Namens an, der in der Option **DESTID** angegeben wird.

PRINT

Gibt an, dass die Ausgabe an ein Druckermedium erfolgt.

SUBADDR(datenwert)

Gibt die Unteradresse des Mediums als Halbwort-Binärwert (im Bereich von 0–15) an, die es ermöglicht, Medien desselben Typs wie zum Beispiel "printer 1" oder "printer 2" zu definieren. Der Wert 15 bedeutet ein Medium eines beliebigen Typs. Der Standardwert ist null.

VOLUME(datenwert)

Gibt den Namen (1 - 6 Zeichen) einer Diskette in einem externen Ziel an, die die Datei enthält, die in der Option **DESTID** angegeben wird.

VOLUMELENG(datenwert)

Gibt die Länge des Namens, der in der Option **VOLUME** angegeben wird, als Halbwort-Binärwert an.

WPMEDIA1 bis WPMEDIA4

Gibt für jede bestimmte Einheit vom LU-Typ 4 (**LUTYPE4**) an, dass ein Textverarbeitungsmedium in Bezug auf eine bestimmte Ein-/Ausabeeinheit definiert ist.

Bedingungen

48 FUNCERR

Wird ausgelöst, wenn ein Fehler bei der Ausführung des Befehls auftritt. Die Zielauswahl bleibt davon unbeeinträchtigt und andere Befehle für dasselbe Ziel werden möglicherweise erfolgreich ausgeführt.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

RESP2-Werte:

200

Eine Serveranwendung für Distributed Program Link hat in der Option **CONVID** die Sitzung für die Funktionsverlagerung (ihre Hauptfunktion) angegeben.

Standardaktion: Die Task wird abnormal beendet.

47 SELNERR

Wird ausgelöst, wenn ein Fehler bei der Zielauswahl auftritt. Das Ziel wird nicht ausgewählt und andere Befehle für dasselbe Ziel werden wahrscheinlich nicht erfolgreich ausgeführt.

Standardaktion: Die Task wird abnormal beendet.

49 UNEXPIN

Tritt auf, wenn nicht erwartete oder nicht erkannte Informationen vom externen Controller empfangen werden.

Standardaktion: Die Task wird abnormal beendet.

JOURNAL

Erstellt einen Journalsatz.

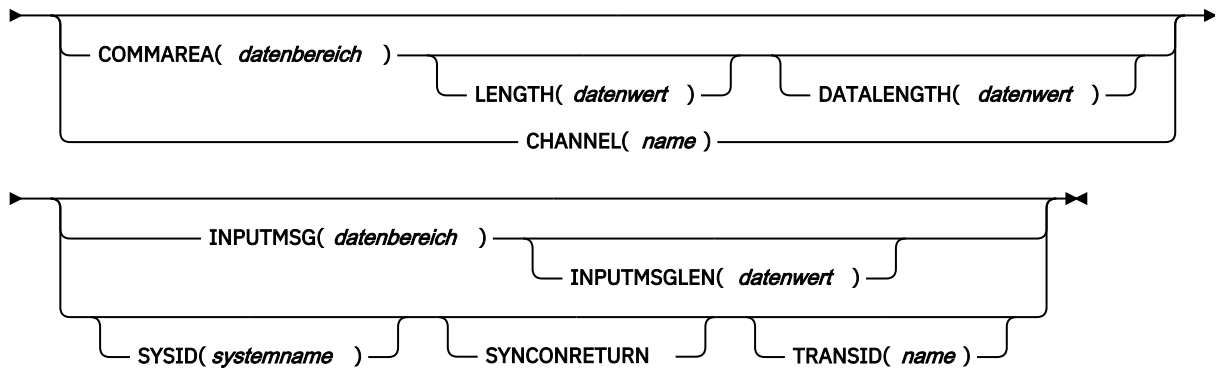
Dieser Befehl wird aus Gründen der Kompatibilität mit früheren Releases von CICS unterstützt. Er wurde durch den Befehl WRITE JOURNALNAME ersetzt, der stattdessen verwendet werden sollte.

LINK

Stellt eine Verknüpfung zu einem anderen Programm her und erwartet eine Rückmeldung.

LINK

➔ LINK — PROGRAM(*name*) ➔



Bedingungen: CHANNELERR, INVREQ, LENGERR, NOTAUTH, PGMIDERR, RESUNAVAIL, ROLLEDBACK, SYSIDERR, TERMERR

Dieser Befehl ist threadsicher, wenn er dazu verwendet wird, ein Programm in einer lokalen CICS-Region oder ein Programm in einer fernen CICS-Region über eine IPIC-Verbindung zu verknüpfen. Er ist nicht threadsicher, wenn er dazu verwendet wird, ein Programm in einer fernen CICS-Region über einen anderen Typ von Verbindung zu verknüpfen.

Beschreibung

Der Befehl LINK übergibt die Steuerung aus einem Anwendungsprogramm auf einer logischen Ebene an ein Anwendungsprogramm auf der nächstniedrigeren logischen Ebene.

Wenn das angeforderte Programm in CICS nicht definiert ist und die automatische Installation (AUTOINSTALL) aktiviert wurde, stellt CICS eine Definition für das Programm bereit. Wenn diese Definition lokal ist und sich das verknüpfte Programm noch nicht im Hauptspeicher befindet, wird es von CICS geladen.

Unter bestimmten Umständen kann sich das verknüpfte Programm in einer anderen CICS-Region befinden (siehe „Distributed Program Link (DPL)” auf Seite 323).

Dieser Befehl wird im aktuellen Anwendungskontext ausgeführt. Wenn der Befehl durch ein Programm abgesetzt wird, das unter einer Task für eine Anwendung ausgeführt wird, die auf einer Plattform bereitgestellt ist, sucht CICS zunächst nach dem genannten Programm im privaten Programmverzeichnis für die Anwendung. Wenn das Programm dort nicht gefunden wird, sucht CICS anschließend im öffentlichen Programmverzeichnis.

Wenn dieser Befehl dazu verwendet wird, eine Verknüpfung zu einem Programm herzustellen, das als Anwendungseinstiegspunkt für eine Anwendung deklariert ist, die auf einer Plattform bereitgestellt wurde,

muss das CICS-Bundle, in dem der Anwendungseinstiegspunkt deklariert ist, den Status AVAILABLE (verfügbar) haben. Die Verknüpfung wird zur Version der Anwendung mit der höchsten Nummer hergestellt, die installiert, aktiviert und verfügbar ist. Wenn eine Verknüpfung zu einer angegebenen Version einer Anwendung, die auf einer Plattform bereitgestellt wurde, hergestellt werden soll, verwenden Sie den Befehl **INVOKE APPLICATION** anstelle des Befehls **LINK**.

Wenn der Befehl **RETURN** in dem verknüpften Programm ausgeführt wird, wird die Steuerung bei der nächsten sequenziellen ausführbaren Anweisung an das Programm zurückgegeben, das die Verknüpfung initiiert hat.

Die externe CICS-Schnittstelle (EXCI) stellt einen Befehl **LINK** bereit, der alle sechs Befehle der Schnittstelle in einem Aufruf ausführt. Informationen zu EXCI finden Sie unter [Schnittstelle EXCI CALL](#).

Das verknüpfte Programm operiert unabhängig von dem Programm, das den Befehl **LINK** absetzt, in Bezug auf die Behandlung von Bedingungen, AID-Zeichen, Abbrüchen und Ausführungsschlüsseln. Zum Beispiel werden die Auswirkungen von Befehlen **HANDLE CONDITION** im verknüpfenden Programm nicht von dem verknüpften Programm übernommen, sondern es werden die ursprünglichen Befehle **HANDLE CONDITION** nach der Rückkehr zum verknüpfenden Programm wiederhergestellt. Weitere Informationen sowie eine Darstellung des Konzepts logischer Ebenen finden Sie unter [Befehl HANDLE CONDITION verwenden](#).

Sie können den Befehl **HANDLE ABEND** verwenden, um abnormale Beendigungen auf anderen Verbindungsebenen zu behandeln. Weitere Informationen zur Beziehung zwischen den Befehlen **LINK** und **HANDLE ABEND** finden Sie unter [Befehl HANDLE CONDITION verwenden](#).

Distributed Program Link (DPL)

In jedem der folgenden Fälle ist die Verknüpfung ein *Distributed Program Link* (DPL):

- Sie geben den Namen einer fernen Region in der Option **SYSID** mit oder ohne zugeordnete Optionen **TRANSID** und **SYNCONRETURN** an.
- Die Option **REMOTESYSTEM** in der installierten Programmdefinition gibt den Namen einer fernen Region an.
- Die installierte Programmdefinition gibt **DYNAMIC(YES)** an oder es gibt keine installierte Programmdefinition und das Programm für dynamisches Routing leitet die **LINK**-Anforderung an eine ferne Region weiter.

Als Antwort auf einen Distributed Program Link sendet die lokale CICS-Region (*Clientregion*) die Verknüpfungsanforderung (**LINK**-Anforderung) an die ferne Region (*Serverregion*). Die Serverregion führt das verknüpfte Programm (Serverprogramm) stellvertretend für das Programm aus, das die **LINK**-Anforderung abgesetzt hat (Clientprogramm).

Die Optionen **SYSID** und **INPUTMSG** schließen sich gegenseitig aus. Wenn Sie beide Optionen in einem Befehl **LINK** angeben, gibt der Umsetzer eine Fehlermeldung DFH7230E aus, die auf eine unzulässige Kombination von Optionen hinweist.

Ein Serverprogramm, das in der Serverregion ausgeführt wird, ist auf eine DPL-Teilmenge der CICS-API beschränkt. Zusammengefasst kann das Serverprogramm die folgenden Befehle nicht absetzen:

- Terminalsteuerungsbefehle, die auf die Hauptfunktion verweisen
- Optionen von **ASSIGN**, die Terminalattribute zurückgeben
- BMS-Befehle
- An- und Abmeldebefehle
- Stapeldatenaustauschbefehle
- Befehle, die den Benutzerbereich der Terminalsteuertabelle (TCTUA - Terminal Control Table User Area) adressieren

Weitere Informationen zur eingeschränkten DPL-Teilmenge der API finden Sie unter [Ausnahmebedingungen für den Befehl LINK](#).

Wenn ein Serverprogramm abgebrochen wird, wird der Abbruchcode an das Clientprogramm zurückgegeben. Falls das Clientprogramm nicht mit einer Behandlung des vom Serverprogramm zurückgegebenen Abbruchcodes geschrieben wurde, wird das Clientprogramm mit demselben Abbruchcode wie dem, der vom Serverprogramm zurückgegeben wurde, abgebrochen.

Sie können DPL nicht zum Herstellen einer Verknüpfung zum CICS-Master-Terminalprogramm DFHEMTA oder zum RDO-Programm DFHEDAP verwenden. Die Adressen, die als Parameter an DFHEMTA und DFHEDAP übergeben werden, sind nur in der Region gültig, die den Befehl EXEC CICS LINK absetzt. Das heißt, dass Sie keine DFHEMTA- oder DFHEDAP-Anforderung an ein fernes CICS weiterleiten können.

Wichtig: Beispiele für die Verwendung des Befehls LINK zur Verknüpfung mit einem fernen Programm finden Sie unter [Distributed Program Link \(DPL\)](#). Informationen zum Schreiben eines Programms für dynamisches Routing finden Sie unter [DPL-Anforderungen dynamisch weiterleiten](#).

Optionen

CHANNEL(name)

Gibt den Namen (1 - 16 Zeichen) eines Kanals an, der für das aufgerufene Programm verfügbar gemacht werden soll. Die folgenden Zeichen sind zulässig: A - Z a - z 0 - 9 \$ @ # / % & ? ! : | " = ~ , ; < > . - und _ . Führende und eingebettete Leerzeichen sind nicht zulässig. Falls der angegebene Name weniger als 16 Zeichen umfasst, wird er mit nachgestellten Leerzeichen auf 16 Zeichen aufgefüllt. Wenn der Kanal nicht vorhanden ist, wird er erstellt. Dieser neue Kanal bleibt im Bereich, bis die Verbindungsebene geändert wird. Informationen zum Geltungsbereich von Kanälen finden Sie unter [Geltungsbereich eines Kanals](#).

Kanalnamen haben immer EBCDIC-Format. Die oben angegebene Gruppe der zulässigen Zeichen für Kanalnamen enthält einige Zeichen, die nicht in allen EBCDIC-Codepages dieselbe Darstellung besitzen. Falls Kanäle zwischen Regionen verlagert werden sollen, empfiehlt es sich daher, bei ihrer Benennung ausschließlich die Zeichen A-Z a-z 0-9 & : = , ; < > . - und _ zu verwenden.

Wenn Sie den Kanalnamen DFHTRANSACTION angeben, wird ein Transaktionskanal erstellt, wenn noch kein Transaktionskanal vorhanden ist. Der Transaktionskanal ist für alle Programme in der aktuellen Task sichtbar. Darüber hinaus wird der Transaktionskanal als aktueller Kanal an das aufgerufene Programm übergeben. Weitere Informationen finden Sie unter [Kanäle und Container](#).

Für das Programm, das den Befehl LINK absetzt, bestehen die folgenden Möglichkeiten:

- Es kann den Kanal durch einen oder mehrere der Befehle **PUT CONTAINER CHANNEL** oder **PUT64 CONTAINER** bereits erstellt haben.
- Es kann seinen aktuellen Kanal durch den Namen angeben.
- Es kann einen Kanal benennen, der zurzeit nicht vorhanden ist. In diesem Fall wird ein neuer leerer Kanal erstellt.

COMMAREA(datenbereich)

Gibt einen Kommunikationsbereich an, der für das aufgerufene Programm verfügbar gemacht werden soll. In dieser Option wird der Datenbereich übergeben und Sie müssen ihm im empfangenden Programm den Namen DFHCOMMAREA geben. COMMAREA wird für einen Befehl LINK an Liberty nicht unterstützt. Siehe [Daten an andere Programme übergeben](#).

DATALENGTH(datenwert)

Gibt einen Halbwort-Binärwert an, der die Länge eines zusammenhängenden Bereichs von Speicher vom Beginn des Kommunikationsbereichs (COMMAREA) an darstellt, der an das aufgerufene Programm übergeben werden soll. Geben Sie DATALENGTH aus Leistungsgründen an, wenn für eine ferne LINK-Anforderung die Menge an Daten, die in einem Kommunikationsbereich (COMMAREA) übergeben wird, klein ist, der Kommunikationsbereich selbst jedoch groß ist, sodass das verknüpfte Programm die angeforderten Daten zurückgeben kann.

Der Wert von DATALENGTH wird nur geprüft, wenn die LINK-Anforderung fern oder dynamisch ist. Für statische lokale Verknüpfungen wird er nicht geprüft.

DATALENGTH kann nicht gleichzeitig mit INPUTMSG verwendet werden.

INPUTMSG(datenbereich)

Gibt Daten an, die für das aufgerufene Programm bereitgestellt werden sollen, wenn es zum ersten Mal einen Befehl RECEIVE absetzt. Die Daten bleiben bis zur Ausführung eines Befehls RECEIVE oder RETURN verfügbar. Ein aufgerufenes Programm kann ein weiteres Programm usw. aufrufen, sodass das eine Kette verknüpfter Programme erstellt wird. Wenn eine verknüpfte Kette vorhanden ist, übergibt CICS die Daten von INPUTMSG an den ersten Befehl RECEIVE, der in der Kette ausgeführt wird. Wenn die Steuerung an das Programm zurückgegeben wird, das den Befehl LINK mit der Option INPUTMSG abgesetzt hat, bevor die Daten von INPUTMSG von einem Befehl RECEIVE empfangen wurden, geht CICS davon aus, dass ein Befehl RECEIVE abgesetzt wurde. In diesem Fall sind die ursprünglichen Daten von INPUTMSG nicht länger verfügbar.

INPUTMSG kann nicht gleichzeitig mit DATALENGTH verwendet werden. INPUTMSG wird für einen Befehl LINK an Liberty nicht unterstützt.

Weitere Informationen zur Option INPUTMSG finden Sie unter [INPUTMSG](#).

INPUTMSGLEN(datenwert)

Gibt einen Halbwort-Binärwert für die Verwendung mit INPUTMSG an.

LENGTH(datenwert)

Gibt einen Halbwort-Binärwert an, der die Länge des Kommunikationsbereichs (COMMAREA) in Byte darstellt.

Wenn die Anforderung ein Distributed Program Link (DPL) zu einer Region mit der Stufe CICS TS V5.5 oder früher oder ein DPL über eine ISC-Verbindung ist, sollte dieser Wert nicht größer als 24 KB sein. Diese Begrenzung berücksichtigt den Kommunikationsbereich und den Platz für Header.

Stellen Sie sicher, dass der Wert, den Sie angeben, der Länge der Daten entspricht, die in dem Kommunikationsbereich übergeben werden. Geben Sie für LENGTH nicht 0 (null) an, weil das daraus resultierende Verhalten unvorhersehbar ist und der Befehl EXEC CICS LINK fehlschlagen könnte.

Wenn Sie einen Kommunikationsbereich zur Übergabe von Daten verwenden, muss das verknüpfte Programm prüfen, ob das Feld EIBCALEN in dem EIB der Task mit dem Wert übereinstimmt, den das Programm erwartet. Abweichungen könnten zu Fehlern im Speicher oder zu Systemfehlern führen. Weitere Informationen finden Sie unter [COMMAREA](#).

PROGRAM(name)

Gibt die ID (1 - 8 Zeichen) des Programms an, an das die Steuerung unbedingt übergeben werden soll.

In jedem der folgenden Fälle ist das verknüpfte Programm ein Serverprogramm in einer fernen Region:

- Die Option SYSID gibt eine ferne Region an.
- Die Option REMOTESYSTEM in der installierten Programmdefinition gibt den Namen einer fernen Region an.
- Die installierte Programmdefinition gibt DYNAMIC(YES) an oder es gibt keine installierte Programmdefinition und das Programm für dynamisches Routing leitet die LINK-Anforderung an eine ferne Region weiter.

Die Verwendung von Anführungszeichen ist zu beachten:

```
EXEC CICS LINK PROGRAM('PROGX')
```

PROGX steht in Anführungszeichen, weil es sich um den Programmnamen handelt.

```
EXEC CICS LINK PROGRAM(DAREA)
```

DAREA steht nicht in Anführungszeichen, weil dies der Name eines Datenbereichs ist, der den tatsächlichen Programmnamen enthält. Wenn für den Programmnamen ein Datenbereich verwendet wird, muss dieser Datenbereich als acht Byte langes Feld im Arbeitsspeicher definiert werden.

Anmerkung: Wenn eine Verknüpfung zu einem CICS 3270-Programm hergestellt wird, das unter dem Link3270-Brückenmechanismus ausgeführt werden soll, muss in PROGRAM der Programmname in PROGRAM DFHL3270 und nicht der Name des 3270-Zielprogramms angegeben werden.

SYNCONRETURN

Gibt an, dass die Serverregion, die in der Option SYSID genannt wird, nach erfolgreichem Abschluss des Serverprogramms einen Synchronisationspunkt erfassen soll.

Änderungen an wiederherstellbaren Ressourcen, die durch das Serverprogramm durchgeführt wurden, werden unabhängig von den Änderungen festgeschrieben oder rückgängig gemacht, die an wiederherstellbaren Ressourcen durch das Clientprogramm durchgeführt wurden, das die LINK-Anforderung abgesetzt hat, oder von Änderungen, die durch den Server in einer beliebigen nachfolgenden Verknüpfung (LINK) durchgeführt wurden.

- Die Bedingung NORMAL wird zurückgegeben, wenn Änderungen an wiederherstellbaren Ressourcen vor der Rückkehr vom Serverprogramm festgeschrieben werden.
- Die Bedingung ROLLEDBACK wird zurückgegeben, wenn Änderungen an wiederherstellbaren Ressourcen vor der Rückkehr vom Serverprogramm rückgängig gemacht werden.
- Die Bedingung TERMERR tritt nach einem Fehler der Kommunikationsverbindung oder des Systems, in dem das Serverprogramm ausgeführt wird, auf. Das Clientprogramm behandelt die Bedingung und stellt sicher, dass die Datenkonsistenz wiederhergestellt wird.

SYNCONRETURN ist nur auf ferne Verknüpfungen (Links) anwendbar und wird bei einer lokalen Verknüpfung ignoriert.

SYSID(systemname)

Gibt den Systemnamen einer CICS-Serverregion an, an die die Programmverknüpfungsanforderung weitergeleitet werden soll.

Wenn Sie in der Option SYSID nicht den Namen eines fernen Systems angeben oder die Option SYSID weglassen, verwendet CICS das Attribut REMOTESYSTEM, das in der installierten Programmdefinition definiert ist. Wenn Sie den Namen eines lokalen Systems in der Option SYSID oder im Attribut REMOTESYSTEM angeben, ignoriert CICS den Namen.

Der Name eines fernen Systems, der in der Option SYSID angegeben wird, hat Vorrang vor jedem Namen eines fernen Systems, der in der Programmressourcendefinition angegeben ist oder durch das Programm für dynamisches Routing zurückgegeben wird.

TRANSID(name)

Gibt den Namen der Spiegeltransaktion an, die von der fernen Region angeschlossen werden soll und unter der sie das Serverprogramm ausführen soll.

Der Transaktionsname, den Sie im Befehl LINK angeben, hat Vorrang vor jeder Transaktion, die in der Programmressourcendefinition angegeben ist. Sie können zwar Ihren eigenen Namen für die Spiegeltransaktion angeben, die durch DPL-Anforderungen initialisiert wird, jedoch muss die Transaktion in der Serverregion definiert sein, und die Transaktionsdefinition muss das Spiegelprogramm DFHMIRS angeben.

Wenn Sie die Option TRANSID nicht angeben, wird auf Programmressourcendefinitionen zurückgegriffen, die lokal gespeichert sind, wenn die installierte Programmdefinition ein fernes Attribut DYNAMIC(YES) angibt. Andernfalls schließt die Serverregion standardmäßig entweder CSMI, CPMI oder CVMi an.

Die Angabe der Option TRANSID mit einem leeren Transaktionsnamen ist nicht gültig. Die Fehlerantwort in diesem Fall hängt davon ab, wie die LINK-Anforderung verarbeitet wird:

- Wenn das verknüpfte Programm lokal definiert ist, wird die LINK-Anforderung in diesem Fall erfolgreich ausgeführt.
- Wenn die Anforderung statisch an eine ferne Region geleitet wird, wird eine Antwort INVREQ an das Programm zurückgegeben, das die Verknüpfung initiiert.
- Wenn die Anforderung dynamisch an eine ferne Region geleitet wird, wird eine gültige Verbindungs-ID zurückgegeben. Nach Prüfung der Transaktions-ID (TRANSID) ruft CICS jedoch das Programm für dynamisches Routing mit dem Wert 1 für den Parameter DYRFUNC und dem Wert 8 für den Parameter DYRERROR erneut auf. Wenn das Programm für dynamisches Routing antwortet, indem es das Programm lokal ausführt, ist die LINK-Anforderung erfolgreich. Wenn das Programm für dynamisches Routing antwortet, indem es den Versuch ohne Erfolg wiederholt, wird die LINK-Anforde-

ung zurückgewiesen und eine Antwort PGMIDERR an das Programm zurückgegeben, das die Verknüpfung initiiert hat.

Bedingungen

122 CHANNELERR

RESP2-Werte:

1

Der in der Option CHANNEL angegebene Name enthält ein ungültiges Zeichen oder eine ungültige Zeichenkombination.

16 INVREQ

RESP2-Werte:

8

Ein Befehl LINK mit der Option INPUTMSG wird für ein Programm abgesetzt, das keinem Terminal zugeordnet ist oder das einer logischen APPC-Einheit oder einer IRC-Sitzung zugeordnet ist.

14

Die Option SYNCONRETURN wurde angegeben, jedoch befindet sich das Programm, das die LINK-Anforderung absetzt (Clientprogramm), bereits im Datenaustausch mit einer Spiegeltask in der fernen Region, die in der Option SYSID angegeben ist. Das heißt, es ist eine Arbeitseinheit (UOW) aktiv oder der Systeminitialisierungsparameter **MROFSE=YES** ist in der Clientregion angegeben oder die Einstellung MIRRORLIFE IPCONN ist als Task oder Arbeitseinheit in der fernen Region angegeben. In diesem Fall befindet sich das Clientprogramm in einem falschen Status für die Unterstützung der Option SYNCONRETURN.

15

Das Programm, das die LINK-Anforderung absetzt, befindet sich bereits im Datenaustausch mit einer Spiegeltask und die angegebene Transaktions-ID (TRANSID) unterscheidet sich von der Transaktions-ID des aktiven Spiegels.

16

Die angegebene Transaktions-ID (TRANSID) besteht ausschließlich aus Leerzeichen.

17

Die Transaktions-ID (TRANSID), die durch das Programm für dynamisches Routing bereitgestellt wurde, besteht nur aus Leerzeichen.

19

Ein Befehl LINK mit der Option INPUTMSG wurde für ein Programm abgesetzt, das Gegenstand einer DPL-Anforderung ist; das heißt, es wurde auch SYSID angegeben.

30

Die Programmmanagerdomäne wurde nicht initialisiert, wahrscheinlich weil eine Verknüpfungsanforderung in einer Programmlistentabelle (PLT) der ersten Stufe ausgeführt wurde.

48

Ein Befehl LINK an ein Java-Programm wurde versucht, jedoch kann die Benutzerklasse nicht gefunden werden.

51

Ein Befehl LINK an ein Java-Programm wurde versucht, jedoch kann die Ressource JVMSERVER nicht gefunden werden.

52

Ein Befehl LINK an ein Java-Programm wurde versucht, jedoch ist die Ressource JVMSERVER nicht aktiviert.

53

Ein Befehl LINK an eine Anwendung in einem Liberty-JVM-Server wurde versucht, jedoch war der LINK-Listenerprozess nicht verfügbar. Die wahrscheinlichste Ursache für diesen Fehler ist, dass das Feature `cicsts:link-1.0` nicht aktiviert ist. Zur Lösung des Problems fügen Sie dieses Feature in der Datei `server.xml` hinzu.

54

Die Option COMMAREA wurde angegeben, ist jedoch ungültig, wenn eine Verknüpfung zu einer Anwendung in einem Liberty-JVM-Server hergestellt wird.

55

Die Option INPUTMSG wurde angegeben, ist jedoch ungültig, wenn eine Verknüpfung zu einer Anwendung in einem Liberty-JVM-Server hergestellt wird.

Standardaktion: Die Task wird abnormal beendet.

Anmerkung: RESP2-Werte werden für Bedingungen, die in einem DPL-Serverprogramm auftreten, nicht an den Client zurückgegeben.

22 LENGERR

RESP2-Werte:

11

Die Länge von COMMAREA ist kleiner als 0 oder größer als die zulässige Länge.

12

Die Länge, die in der Option DATALENGTH angegeben wurde, ist ein negativer Wert.

13

Die Länge, die in der Option DATALENGTH angegeben wurde, ist größer als die in der Option LENGTH angegebene Länge.

26

Die Adresse von COMMAREA ist null, jedoch ist die Länge von COMMAREA ungleich null.

27

Die Länge von INPUTMSG ist kleiner 0 oder größer als 32767.

Tritt außerdem in einer der folgenden Situationen auf (RESP2 nicht festgelegt):

- Die Länge, die in der Option LENGTH angegeben wurde, ist größer als die Länge des Datenbereichs, der in der Option COMMAREA angegeben wurde, und beim Kopieren dieser Daten ist wegen der falschen Längenangabe eine zerstörerische Überlappung aufgetreten.

Standardaktion: Die Task wird abnormal beendet.

Anmerkung: RESP2-Werte werden für Bedingungen, die in einem DPL-Serverprogramm auftreten, nicht an den Client zurückgegeben.

70 NOTAUTH

RESP2-Werte:

101

Eine Ressourcensicherheitsprüfung ist für PROGRAM(name) fehlgeschlagen.

Standardaktion: Die Task wird abnormal beendet.

27 PGMIDERR

RESP2-Werte:

1

Ein Programm hat keine installierte Ressourcendefinition und die automatische Programminstallation war inaktiviert oder das Steuerprogramm zur automatischen Programminstallation gab an, dass das Programm nicht automatisch installiert werden sollte.

Dieser RESP2-Code kann auch auftreten, wenn das Programm ein privates Programm für eine Anwendung ist, das nicht als Einstiegspunkt definiert ist, und der aktuelle Anwendungskontext für Tasks nicht zulässt, dass es aufgerufen wird.

2

Ein Programm ist inaktiviert.

3

Ein Programm wurde aus folgenden Gründen nicht geladen:

- Dieses Laden des Programms war das erste Laden und das Programmladen ist fehlgeschlagen, in der Regel weil das Lademodul nicht gefunden wurde.
- Dieses Laden des Programms war ein nachfolgendes Laden, jedoch ist das erste Laden fehlgeschlagen.

Zum Zurücksetzen des Ladestatus muss sich das Lademodul in der CICS-Programmbibliotheksverkettung (DFHRPL-Verkettung) befinden und ein Befehl SET PROGRAM NEWCOPY ist erforderlich.

21

Das Steuerprogramm zur automatischen Programminstallation ist fehlgeschlagen, entweder weil das Steuerprogramm zur automatischen Programminstallation falsch ist, weil es falsch definiert ist oder weil ein Abbruch im Steuerprogramm zur automatischen Programminstallation aufgetreten ist. Die automatische Programminstallation wird inaktiviert und die Nachricht DFHPG0202 oder DFHPG0203 wird in den CSPL geschrieben.

22

Das Modell, das von dem Steuerprogramm zur automatischen Programminstallation zurückgegeben wurde, war in CICS nicht definiert oder war nicht aktiviert.

23

Das Steuerprogramm zur automatischen Programminstallation hat ungültige Daten zurückgegeben.

24

Die Definition für das Programm ist fehlgeschlagen, weil die automatische Installation einen ungültigen Programmnamen oder eine ungültige Definition zurückgegeben hat.

25

Das Programm für dynamisches Routing hat die LINK-Anforderung zurückgewiesen.

26

Ein Befehl LINK an eine Anwendung in einem Liberty-JVM-Server wurde versucht, jedoch ist die Anforderung fehlgeschlagen und die Zielanwendung wurde nicht aufgerufen. Die Nachricht DFHSJ1207 gibt weitere Details zu den Ursachen an, aus denen die Anwendung nicht aufgerufen werden konnte.

27

Ein Befehl LINK an eine Anwendung in einem Liberty-JVM-Server wurde versucht, jedoch hat die Anforderung das Zeitlimit überschritten und die Zielanwendung wurde nicht aufgerufen. Weitere Informationen finden Sie unter [Fehlerbehebung bei Liberty-JVM-Servern und Java-Webanwendungen](#).

Standardaktion: Die Task wird abnormal beendet.

Anmerkung: RESP2-Werte werden für Bedingungen, die in einem DPL-Serverprogramm auftreten, nicht an den Client zurückgegeben.

121 RESUNAVAIL

RESP2-Werte:

0

Eine Ressource, die für das verknüpfte Programm erforderlich ist, ist in der Zielregion nicht verfügbar. Die Bedingung RESUNAVAIL gilt für dynamisch weitergeleitete DPL-Anforderungen (DPL - Distributed Program Link).

RESUNAVAIL wird für den Befehl **EXEC CICS LINK** zurückgegeben, der *von dem Spiegel in der Zielregion ausgeführt wird*, wenn ein globales Benutzerexitprogramm XPCERES angibt, dass eine erforderliche Ressource in der Zielregion nicht verfügbar ist. Diese Bedingung wird nicht an die Anwendung zurückgegeben.

Standardaktion: Das Programm für dynamisches Routing wird wegen des Routenauswahlfehlers erneut aufgerufen.

82 ROLLEDBACK

RESP2-Werte:

29

Die Option SYNCONRETURN wurde angegeben und das Serverprogramm kann einen Synchronisationspunkt nicht erfolgreich erfassen. Das Serverprogramm hat einen Rollback durchgeführt und alle Änderungen, die in der aktuellen Arbeitseinheit (UOW) an wiederherstellbaren Ressourcen in der fernen Region vorgenommen wurden, wurden zurückgesetzt.

Standardaktion: Die Task wird abnormal beendet.

53 SYSIDERR

RESP2-Werte:

18

Die angegebene System-ID (SYSID) konnte in der systemübergreifenden Tabelle nicht gefunden werden.

20

Das ferne System, das durch SYSID angegeben wurde, ist ein durch LUTYPE6.1 verbundenes System. DPL-Anforderungen werden über LUTYPE6.1-Verbindungen nicht unterstützt.

Anmerkung:

1. Es erfolgt keine lokale Warteschlangensteuerung im Fall einer Bedingung SYSIDERR.
2. Es werden keine RESP2-Werte für Bedingungen zurückgegeben, die bei DPL-Anforderungen auftreten.

21

Die Option CHANNEL wurde verwendet und die LINK-Anforderung wurde an ein fernes System gesendet oder weitergeleitet, das diese Option nicht unterstützt. (IPIC- und MRO-Verbindungen.)

28

Das ferne System, das durch SYSID angegeben wurde, ist nicht aktiv. Diese Antwort kann auch darauf hinweisen, dass die Transaktion auf dem fernen System nicht definiert wurde.

29

Das ferne System, das durch SYSID angegeben wurde, ist aktiv, jedoch sind keine Sitzungen verfügbar und das Programm für dynamisches Routing hat entschieden, die LINK-Anforderung nicht in die Warteschlange zu stellen.

31

Die Anforderung zum Zuordnen einer Sitzung an das ferne System wurde zurückgewiesen.

32

In der Warteschlange für Zuordnungsanforderungen für Sitzungen an das ferne System ist ein Fehler aufgetreten, weil die Warteschlange für Sitzungszuordnungen voll ist oder bereinigt wurde.

Standardaktion: Die Task wird abnormal beendet.

81 TERMERR

RESP2-Werte:

17

Beim Datenaustausch mit dem Spiegel tritt ein nicht behebbarer Fehler auf, zum Beispiel dadurch, dass die Sitzung einen Fehler verursacht oder dass die Serverregion ausfällt.

Standardaktion: Die Task wird abnormal beendet.

Wenn im Befehl LINK die Option SYNCONRETURN nicht angegeben wurde, muss das Clientprogramm entscheiden, ob nach Empfang dieser Bedingung ein Abbruch oder ein Rollback erfolgen soll.

Anmerkung: RESP2-Werte werden für Bedingungen, die in einem DPL-Serverprogramm auftreten, nicht an den Client zurückgegeben.

Beispiele

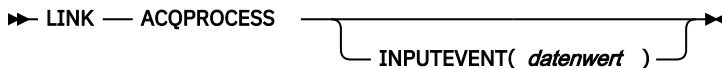
Das folgende Beispiel zeigt, wie eine Verknüpfung (Link) zu einem Anwendungsprogramm mit dem Namen PROGRAM angefordert wird:

```
EXEC CICS LINK PROGRAM(PROGNAME)
      COMMAREA(COMA) LENGTH(LENA)
      DATALENGTH(LENI) SYSID('CONX')
```

LINK ACQPROCESS

Führt einen CICS-BTS-Prozess synchron ohne Kontextwechsel aus.

LINK ACQPROCESS



Bedingungen: EVENTERR, INVREQ, IOERR, NOTAUTH, PGMIDERR, PROCESSBUSY, PROCESSERR

Beschreibung

Der Befehl LINK ACQPROCESS führt den CICS-BTS-Prozess aus, der zurzeit durch den Requester angefordert ist. Der Prozess wird synchron zum Requester (Anforderer) ohne Kontextwechsel ausgeführt.

Der einzige Prozess, den ein Programm durch LINK verknüpfen kann, ist derjenige, den es in der aktuellen Arbeitseinheit angefordert hat. (Beachten Sie jedoch, dass das Programm, wenn es als Aktivierung einer Aktivität ausgeführt wird, einen Befehl RUN und nicht LINK zum Aktivieren des Prozesses verwenden muss, den es angefordert hat.) Siehe [Prozesse und Aktivitäten anfordern](#).

Zur Prüfung der Antwort von dem Prozess muss der Befehl CHECK ACQPROCESS verwendet werden. Dies hat den Grund, dass die Antwort auf die Anforderung zum Aktivieren des Prozesses keine Informationen über Erfolg oder Fehler des Prozesses selbst enthält, sondern nur über Erfolg oder Fehler der Anforderung zum Aktivieren des Prozesses. In der Regel wird der Befehl CHECK sofort nach dem Befehl LINK abgesetzt.

Der Befehl LINK ACQPROCESS veranlasst BTS, die Stammaktivität des Prozesses aufzurufen und dieser ein Eingabeereignis zu senden. Wenn sich die Stammaktivität in ihrem Anfangsstatus befindet, d. h., wenn dies das erste Mal ist, dass sie ausgeführt werden soll, sendet CICS das Systemereignis DFHINITIAL an die Stammaktivität. Befindet sich die Stammaktivität nicht in ihrem Anfangsstatus, muss das Eingabeereignis in der Option INPUTEVENT angegeben werden.

Kein Kontextwechsel

Wenn ein Prozess durch einen Befehl LINK ACQPROCESS aktiviert wird, wird er synchron zum Anforderer wie folgt aufgerufen:

- Er wird in derselben Arbeitseinheit wie der Anforderer aufgerufen.
- Er wird mit den Transaktionsattributen (TRANSID und USERID) der anfordernden Transaktion aufgerufen.

Mit anderen Worten, es findet kein Kontextwechsel statt. Zum asynchronen Aufrufen eines Prozesses *mit* Kontextwechsel, d. h. in einer von der anfordernden Transaktion separaten Arbeitseinheit (UOW) und mit Angabe der Attribute TRANSID und USERID, die im entsprechenden Befehl DEFINE PROCESS angegeben wurden, verwenden Sie den Befehl RUN ACQPROCESS SYNCHRONOUS.

Anmerkung: Ein Kontextwechsel findet immer statt, wenn ein Prozess asynchron ausgeführt wird.

Wenn die Leistung wichtiger als Fehlerisolation, Wiederherstellbarkeit und Sicherheit ist, verwenden Sie den Befehl LINK ACQPROCESS anstelle des Befehls RUN ACQPROCESS SYNCHRONOUS.

Optionen

ACQPROCESS

Gibt an, dass der Prozess, der zurzeit durch den Requester angefordert ist, ausgeführt werden soll.

INPUTEVENT(datenwert)

Gibt den Namen (1-16 Zeichen) des Ereignisses an, das veranlasst, dass der Prozess angeschlossen wird.

Sie dürfen diese Option *nicht* angeben, wenn sich die Stammaktivität des Prozesses in ihrem Anfangsstatus (INITIAL) befindet, d. h., wenn dies das erste Mal ist, dass der Prozess ausgeführt werden soll. In diesem Fall sendet CICS der Stammaktivität das Systemereignis DFHINITIAL.

Sie *müssen* diese Option angeben, wenn sich die Stammaktivität nicht in ihrem Anfangsstatus befindet, d. h. wenn sie zuvor bereits aktiviert wurde.

Wenn Sie die Option INPUTEVENT angeben, muss für eine erfolgreiche Ausführung des Befehls LINK das angegebene Ereignis für die Stammaktivität als Eingabeereignis definiert sein.

Bedingungen

111 EVENTERR

RESP2-Werte:

7

Das in der Option INPUTEVENT angegebene Ereignis wurde von der Stammaktivität des auszuführenden Prozesses nicht als Eingabeereignis definiert oder der Ausführungsstatus des Ereignisses ist FIRED (ausgelöst).

16 INVREQ

RESP2-Werte:

15

Die Task, die den Befehl LINK abgesetzt hat, hat keinen Prozess definiert oder abgerufen.

23

Der Prozess ist ausgesetzt und kann daher nicht synchron ausgeführt werden.

40

Das Programm, das den auszuführenden Prozess implementiert, ist ein fernes Programm.

44

Ein Befehl LINK an ein Java-Programm wurde versucht, jedoch ist der JVM-Pool inaktiviert.

45

Ein Befehl LINK an ein Java-Programm wurde versucht, jedoch kann das JVM-Profil nicht gefunden werden.

46

Ein Befehl LINK an ein Java-Programm wurde versucht, jedoch ist das JVM-Profil nicht gültig.

47

Ein Befehl LINK an ein Java-Programm wurde versucht, jedoch kann die Systemeigenschaftendatei nicht gefunden werden.

48

Ein Befehl LINK an ein Java-Programm wurde versucht, jedoch kann die Benutzerklasse nicht gefunden werden.

49

Der Cache für gemeinsam genutzte Klassen ist gestoppt und das automatische Starten ist inaktiviert, sodass ein Java-Programm, das die Verwendung des Cache für gemeinsam genutzte Klassen anfordert, nicht ausgeführt werden kann.

17 IOERR

RESP2-Werte:

29

Die Repository-Datei ist nicht verfügbar.

30

Es ist ein Ein-/Ausgabefehler für die Repository-Datei aufgetreten.

70 NOTAUTH

RESP2-Werte:

101

Der Benutzer, der der absetzenden Task zugeordnet ist, ist nicht berechtigt, den Prozess auszuführen.

27 PGMIDERR

RESP2-Werte:

1

Ein Programm hat keine installierte Ressourcendefinition und die automatische Programminstallation war inaktiviert oder das Benutzerprogramm zur automatischen Programminstallation gab an, dass das Programm nicht automatisch installiert werden sollte.

2

Ein Programm ist inaktiviert.

3

Ein Programm konnte aus folgenden Gründen nicht geladen werden:

- Dies war das erste Laden des Programms und das Programmladen ist fehlgeschlagen, in der Regel weil das Lademodul nicht gefunden werden konnte.
- Dies war ein nachfolgendes Laden des Programms, jedoch ist das erste Laden fehlgeschlagen.

Zum Zurücksetzen des Ladestatus muss sich das Lademodul in der CICS-Programmbibliothek (DFHRPL) oder in der dynamischen Bibliotheksverkettung (LIBRARY) befinden und ein Befehl SET PROGRAM NEWCOPY ist erforderlich.

21

Das Benutzerprogramm zur automatischen Programminstallation ist fehlgeschlagen, entweder weil das Benutzerprogramm zur automatischen Programminstallation falsch ist, weil es falsch definiert ist oder weil ein Abbruch im Benutzerprogramm zur automatischen Programminstallation aufgetreten ist. Die automatische Programminstallation wird inaktiviert und die Nachricht DFHPG0202 oder DFHPG0203 wird in den CSPL geschrieben.

22

Das Modell, das von dem Benutzerprogramm zur automatischen Programminstallation zurückgegeben wurde, war für CICS nicht definiert oder war nicht aktiviert.

23

Das Benutzerprogramm zur automatischen Programminstallation hat ungültige Daten zurückgegeben.

24

Das Definieren für das Programm ist fehlgeschlagen, weil die automatische Installation einen ungültigen Programmnamen oder eine ungültige Definition zurückgegeben hat.

106 PROCESSBUSY

RESP2-Werte:

13

Die Anforderung hat das Zeitlimit überschritten. Es kann sein, dass die Beendigung einer anderen Task, die diesen Prozessdatensatz verwendet, verhindert wurde.

108 PROCESSERR

RESP2-Werte:

6

Ein anderer Prozess ist aktuell. Das heißt, das Programm, das den Befehl LINK abgesetzt hat, kann keine Verknüpfung zu dem Prozess herstellen, den es angefordert hat, weil es selbst als Aktivierung eines Prozesses ausgeführt wird.

9

Der Prozesstyp konnte nicht gefunden werden.

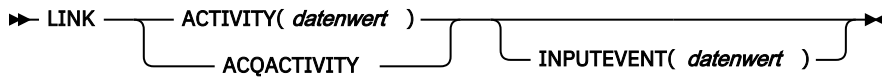
14

Die Stammaktivität des Prozesses, der ausgeführt werden soll, ist nicht im Modus INITIAL oder DORMANT.

LINK ACTIVITY

Führt eine CICS-BTS-Aktivität synchron ohne Kontextwechsel aus.

LINK ACTIVITY



Bedingungen: ACTIVITYBUSY, ACTIVITYERR, EVENTERR, INVREQ, IOERR, LOCKED, NOTAUTH, PGMI-DERR

Beschreibung

Der Befehl LINK ACTIVITY führt eine CICS-BTS-Aktivität (BTS - Business Transaction Services) synchron zum Anforderer (Requester) ohne Kontextwechsel aus. Die Aktivität muss zuvor für BTS definiert worden sein.

Der Befehl LINK ACTIVITY veranlasst BTS, die Aktivität aufzurufen und dieser ein Eingabeereignis zu senden. Wenn sich die Aktivität in ihrem Anfangsstatus befindet, d. h., wenn dies das erste Mal ist, dass sie ausgeführt werden soll oder wenn sie durch einen Befehl RESET ACTIVITY zurückgesetzt wurde, sendet CICS das Systemereignis DFHINITIAL an die Aktivität. Befindet sich die Aktivität nicht in ihrem Anfangsstatus, muss das Eingabeereignis in der Option INPUTEVENT angegeben werden.

Die einzigen Aktivitäten, zu denen ein Programm eine Verknüpfung (LINK) herstellen kann, sind die folgenden:

- Seine eigenen untergeordneten Aktivitäten, falls es als Aktivierung einer Aktivität ausgeführt wird. Es kann mehrere der eigenen untergeordneten Aktivitäten innerhalb derselben Arbeitseinheit verknüpfen.
- Die Aktivität, die es durch einen Befehl ACQUIRE ACTIVITYID in der aktuellen Arbeitseinheit angefordert hat. (Beachten Sie jedoch, dass das Programm, wenn es als Aktivierung einer Aktivität ausgeführt wird, einen Befehl RUN und nicht LINK zum Aktivieren der Aktivität verwenden muss, die es angefordert hat.)

Zur Prüfung der Antwort von der Aktivität muss der Befehl CHECK ACTIVITY verwendet werden. Dies hat den Grund, dass die Antwort auf die Anforderung zum Aktivieren der Aktivität keine Informationen über Erfolg oder Fehler der Aktivität selbst enthält, sondern nur über Erfolg oder Fehler der Anforderung zum Aktivieren der Aktivität. In der Regel wird der Befehl CHECK sofort nach dem Befehl LINK abgesetzt.

Kein Kontextwechsel

Wenn eine Aktivität durch einen Befehl LINK ACTIVITY aktiviert wird, wird sie synchron zum Anforderer wie folgt aufgerufen:

- Sie wird in derselben Arbeitseinheit wie der Anforderer aufgerufen.
- Sie wird mit den Transaktionsattributen (TRANSID und USERID) der anfordernden Transaktion aufgerufen.

Mit anderen Worten, es findet kein **Kontextwechsel** statt. Zum asynchronen Aufrufen einer Aktivität *mit* Kontextwechsel, d. h. in einer von der anfordernden Transaktion separaten Arbeitseinheit (UOW) und mit

Angabe der Attribute TRANSID und USERID, die im entsprechenden Befehl DEFINE ACTIVITY angegeben wurden, verwenden Sie den Befehl RUN ACTIVITY SYNCHRONOUS.

Anmerkung: Ein Kontextwechsel findet immer statt, wenn eine Aktivität asynchron ausgeführt wird.

Wenn die Leistung wichtiger als Fehlerisolation, Wiederherstellbarkeit und Sicherheit ist, verwenden Sie den Befehl LINK ACTIVITY anstelle des Befehls RUN ACTIVITY SYNCHRONOUS.

Optionen

ACQACTIVITY

Gibt an, dass die auszuführende Aktivität diejenige ist, die von der aktuellen Arbeitseinheit durch einen Befehl ACQUIRE ACTIVITYID angefordert wurde.

ACTIVITY(datenwert)

Gibt den Namen (1-16 Zeichen) der Aktivität an, die ausgeführt werden soll. Der Name muss der einer untergeordneten Aktivität der aktuellen Aktivität sein.

INPUTEVENT(datenwert)

Gibt den Namen (1-16 Zeichen) des Ereignisses an, das veranlasst, dass die Aktivität angeschlossen (aktiviert) wird.

Sie dürfen diese Option *nicht* angeben, wenn sich die Aktivität in ihrem Anfangsstatus (INITIAL) befindet, d. h., wenn dies das erste Mal ist, dass sie ausgeführt werden soll, oder wenn sie durch einen Befehl RESET ACTIVITY zurückgesetzt wurde. In diesem Fall sendet CICS der Aktivität das Systemereignis DFHINITIAL.

Sie *müssen* diese Option angeben, wenn sich die Aktivität nicht in ihrem Anfangsstatus befindet, d. h. wenn sie zuvor bereits aktiviert wurde und nicht durch einen Befehl RESET ACTIVITY zurückgesetzt wurde.

Wenn Sie die Option INPUTEVENT angeben, muss für eine erfolgreiche Ausführung des Befehls LINK das angegebene Ereignis für die anzuschließende Aktivität als Eingabeereignis definiert sein.

Bedingungen

107 ACTIVITYBUSY

RESP2-Werte:

19

Die Anforderung hat das Zeitlimit überschritten. Es kann sein, dass die Beendigung einer anderen Task, die diesen Aktivitätsdatensatz verwendet, verhindert wurde.

109 ACTIVITYERR

RESP2-Werte:

8

Die in der Option ACTIVITY angegebene Aktivität konnte nicht gefunden werden.

14

Die Zielaktivität befindet sich nicht im korrekten Modus, um die angegebene Ereignisoption zu verarbeiten. Wenn die Option INPUTEVENT nicht angegeben wurde, muss sich die Aktivität im Modus INITIAL (Anfangsmodus) befinden. Wenn die Option INPUTEVENT angegeben wurde, muss sich die Aktivität im Modus DORMANT (Ruhemodus) befinden.

111 EVENTERR

RESP2-Werte:

7

Das in der Option INPUTEVENT angegebene Ereignis wurde von der auszuführenden Aktivität nicht als Eingabeereignis definiert oder der Ausführungsstatus des Ereignisses ist FIRED (ausgelöst).

16 INVREQ

RESP2-Werte:

4

Mit der Option ACTIVITY wurde eine untergeordnete Aktivität angegeben, aber der Befehl wurde außerhalb des Geltungsbereichs einer gegenwärtig aktiven Aktivität ausgegeben.

21

Die Aktivität ist ausgesetzt und kann daher nicht synchron ausgeführt werden.

24

Die Option ACQACTIVITY wurde verwendet, die absetzende Task hat jedoch keine Aktivität angefordert.

40

Das Programm, das die Aktivität implementiert, ist ein fernes Programm.

44

Ein Befehl LINK an ein Java-Programm wurde versucht, jedoch ist der JVM-Pool inaktiviert.

45

Ein Befehl LINK an ein Java-Programm wurde versucht, jedoch kann das JVM-Profil nicht gefunden werden.

46

Ein Befehl LINK an ein Java-Programm wurde versucht, jedoch ist das JVM-Profil nicht gültig.

47

Ein Befehl LINK an ein Java-Programm wurde versucht, jedoch kann die Systemeigenschaftendatei nicht gefunden werden.

48

Ein Befehl LINK an ein Java-Programm wurde versucht, jedoch kann die Benutzerklasse nicht gefunden werden.

49

Der Cache für gemeinsam genutzte Klassen ist gestoppt und das automatische Starten ist inaktiviert, sodass ein Java-Programm, das die Verwendung des Cache für gemeinsam genutzte Klassen anfordert, nicht ausgeführt werden kann.

17 IOERR

RESP2-Werte:

29

Die Repository-Datei ist nicht verfügbar.

30

Es ist ein Ein-/Ausgabefehler für die Repository-Datei aufgetreten.

100 LOCKED

Die Anforderung kann nicht ausgeführt werden, weil eine Sperre für den relevanten Datensatz in der Repository-Datei besteht.

70 NOTAUTH

RESP2-Werte:

101

Der Benutzer, der der absetzenden Task zugeordnet ist, ist nicht berechtigt, die Aktivität auszuführen.

27 PGMIDERR

RESP2-Werte:

1

Ein Programm hat keine installierte Ressourcendefinition und die automatische Programminstallation war inaktiviert oder das Benutzerprogramm zur automatischen Programminstallation gab an, dass das Programm nicht automatisch installiert werden sollte.

2

Ein Programm ist inaktiviert.

3

Ein Programm konnte aus folgenden Gründen nicht geladen werden:

- Dies war das erste Laden des Programms und das Programmladen ist fehlgeschlagen, in der Regel weil das Lademodul nicht gefunden werden konnte.
- Dies war ein nachfolgendes Laden des Programms, jedoch ist das erste Laden fehlgeschlagen.

Zum Zurücksetzen des Ladestatus muss sich das Lademodul in der CICS-Programmbibliothek (DFHRPL) oder in der dynamischen Bibliotheksverkettung (LIBRARY) befinden und ein Befehl SET PROGRAM NEWCOPY ist erforderlich.

21

Das Benutzerprogramm zur automatischen Programminstallation ist fehlgeschlagen, entweder weil das Benutzerprogramm zur automatischen Programminstallation falsch ist, weil es falsch definiert ist oder weil ein Abbruch im Benutzerprogramm zur automatischen Programminstallation aufgetreten ist. Die automatische Programminstallation wird inaktiviert und die Nachricht DFHPG0202 oder DFHPG0203 wird in den CSPL geschrieben.

22

Das Modell, das von dem Benutzerprogramm zur automatischen Programminstallation zurückgegeben wurde, war für CICS nicht definiert oder war nicht aktiviert.

23

Das Benutzerprogramm zur automatischen Programminstallation hat ungültige Daten zurückgegeben.

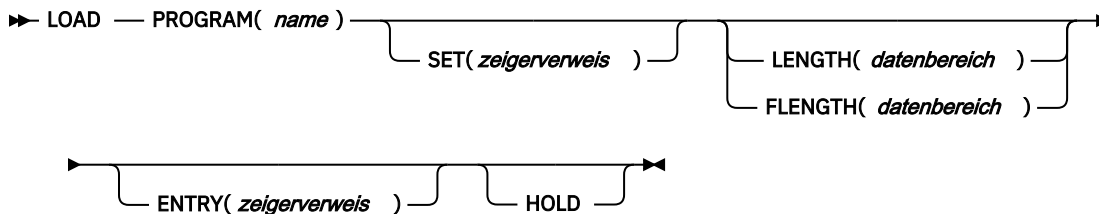
24

Das Definieren für das Programm ist fehlgeschlagen, weil die automatische Installation einen ungültigen Programmnamen oder eine ungültige Definition zurückgegeben hat.

LOAD

Lädt ein Programm aus der CICS-Programmbibliothek (DFHRPL) oder der dynamischen Bibliotheksverkettung (LIBRARY) in den Hauptspeicher.

LOAD



Bedingungen: INVREQ, LENGERR, NOTAUTH, PGMIDERR

Dieser Befehl ist threadsicher.

Hinweis für dynamisches Transaktionsrouting: Die Verwendung von LOAD mit der Option HOLD oder die Verwendung einer Ressource, die mit RELOAD=YES definiert ist, könnte transaktionsübergreifende Affinitäten erstellen, die sich negativ auf die Verwendung des dynamischen Transaktionsroutings auswirken. Weitere Informationen zu Transaktionsaffinitäten finden Sie unter [Affinität](#).

Beschreibung

Der Befehl LOAD erstellt eine Kopie eines Anwendungsprogramms, einer Tabelle oder einer Maske, die für die aufrufende Task verfügbar ist. Wenn das Programm mit RELOAD=NO definiert ist, wird es aus der Bibliotheksverkettung abgerufen, in der es sich nur befindet, wenn keine Kopie bereits im Hauptspeicher vorhanden ist. Wenn das Programm mit RELOAD=YES definiert ist, wird immer eine neue Kopie aus der Bibliotheksverkettung abgerufen. (Siehe weitere Einzelheiten zu Masken unter .) Die Verwendung von LOAD kann den Systemaufwand verringern.

Optionen

ENTRY(*zeigerverweis*)

Gibt den Zeigerverweis an, der auf die Adresse des Einstiegspunkts in dem Programm gesetzt werden soll, das geladen wurde. CICS-Programmladeservices legen den Einstiegspunkt entsprechend dem Adressierungsmodus (AMODE) des Lademoduls fest:

- AMODE(24): Bit 0 ist 0 und Bit 31 ist 0.
- AMODE(31): Bit 0 ist 1 und Bit 31 ist 0.
- AMODE(64): Bit 0 ist 0 und Bit 31 ist 1.

Für Assemblerprogramme, bei denen die Option ENTRY nicht explizit in den Programmverbindungsdefinitionen definiert ist, hängt der Einstiegspunkt, der zurückgegeben wird, davon ab, ob ein CICS-Stub vorhanden ist, sowie davon, ob der Befehl LOAD in einem PLT-Programm abgesetzt wird.

- Wenn ein CICS-Stub vorhanden ist, wird die Einstiegspunktadresse für diesen Stub inkrementiert, sofern der Befehl LOAD nicht aus einem PLT-Programm abgesetzt wurde, das während der ersten Phase der Initialisierung oder der letzten Phase des Herunterfahrens ausgeführt wurde.
- Wenn kein CICS-Stub vorhanden ist, stimmt die Einstiegspunktadresse mit der Ladepunktadresse überein.

FLENGTH(*datenbereich*)

Gibt einen Vollwort-Binärwert-Datenbereich an, der auf die Länge des geladenen Programms, der geladenen Tabelle oder der geladenen Maske gesetzt werden soll. Verwenden Sie die Option FLENGTH, wenn die Länge des geladenen Programms 32 KB übersteigt.

HOLD

Gibt an, dass das geladene Programm, die geladene Tabelle oder die geladene Maske nicht freigegeben werden soll (sofern noch verfügbar), wenn die Task, die den Befehl LOAD absetzt, beendet wird. Die Freigabe soll nur als Antwort auf einen Befehl RELEASE von dieser oder einer anderen Task erfolgen.

Wenn Sie die Option HOLD nicht angeben, wird das Programm, die Tabelle bzw. die Maske freigegeben, wenn die Task, die den Befehl LOAD abgesetzt hat, beendet wird oder einen Befehl RELEASE absetzt.

Wenn das Programm jedoch mit RELOAD=YES definiert wurde, trifft keiner der oben genannten Fälle zu. Der Befehl RELEASE funktioniert nicht und eine Anforderung FREEMAIN muss abgesetzt werden, um das Programm zu entfernen.

LENGTH(*datenbereich*)

Gibt einen Halbwort-Binärwert an, der auf die Länge des geladenen Programms, der geladenen Tabelle oder der geladenen Maske gesetzt werden soll. Verwenden Sie zur Vermeidung der Bedingung LENGERR die Option FLENGTH, wenn das geladene Programm die Länge von 32 KB wahrscheinlich überschreitet.

PROGRAM(*name*)

Gibt die ID (1 - 8 Zeichen) des Programms, der Tabelle oder der Maske an, das bzw. die geladen werden soll. Der angegebene Name muss als Programm in CICS definiert worden sein, obwohl bei aktiver automatischer Installation (AUTOINSTALL) eine Definition automatisch installiert wird.

SET(*zeigerverweis*)

Gibt den Zeigerverweis an, der auf die Adresse gesetzt werden soll, an der ein Programm, eine Tabelle oder eine Maske geladen wurde.

Bedingungen

16 INVREQ

RESP2-Werte:

30

Die Programmmanagerdomäne ist noch nicht initialisiert. Dies ist wahrscheinlich darauf zurückzuführen, dass eine Ladeanforderung in einer PLT der ersten Stufe ausgeführt wurde.

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

RESP2-Werte:

19

Die Option LENGTH wurde verwendet und die Länge des geladenen Programms überschreitet 32 KB.

Standardaktion: Die Task wird abnormal beendet.

70 NOTAUTH

RESP2-Werte:

101

Eine Ressourcensicherheitsprüfung ist für PROGRAM(name) fehlgeschlagen.

Standardaktion: Die Task wird abnormal beendet.

27 PGMIDERR

RESP2-Werte:

1

Ein Programm, eine Tabelle oder eine Maske hat keine installierte Ressourcendefinition und die automatische Programminstallation war inaktiviert oder das Steuerprogramm zur automatischen Programminstallation gab an, dass das Programm nicht automatisch installiert werden sollte.

2

Ein Programm ist inaktiviert.

3

Ein Programm konnte aus einer der folgenden Ursachen nicht geladen werden:

- Dies war das erste Laden des Programms und das Programmladen ist fehlgeschlagen, in der Regel weil das Lademodul nicht gefunden werden konnte.
- Dies war ein nachfolgendes Laden des Programms, jedoch ist das erste Laden fehlgeschlagen.

Zum Zurücksetzen des Ladestatus muss sich das Lademodul in der CICS-Programmbibliothek (DFHRPL) oder in der dynamischen Bibliotheksverkettung (LIBRARY) befinden und ein Befehl SET PROGRAM NEWCOPY ist erforderlich.

9

Die installierte Programmdefinition bezieht sich auf ein fernes Programm.

21

Das Steuerprogramm zur automatischen Programminstallation ist fehlgeschlagen, entweder weil das Steuerprogramm zur automatischen Programminstallation falsch ist, weil es falsch definiert ist oder weil ein Abbruch im Steuerprogramm zur automatischen Programminstallation aufgetreten ist. Die automatische Programminstallation wird inaktiviert und die Nachricht DFHPG0202 oder DFHPG0203 wird in den CSPL geschrieben.

22

Das Modell, das von dem Steuerprogramm zur automatischen Programminstallation zurückgegeben wurde, war in CICS nicht definiert oder war nicht aktiviert.

23

Das Steuerprogramm zur automatischen Programminstallation hat ungültige Daten zurückgegeben.

24

Das Definieren für das Programm ist fehlgeschlagen, weil die automatische Installation einen ungültigen Programmnamen oder eine ungültige Definition zurückgegeben hat.

42

Es wurde versucht, ein JVM-Programm zu laden. Diese Aktion ist nicht gültig, weil Java-Bytecodeprogramme nicht vom CICS Ladeprogramm verwaltet werden.

Standardaktion: Die Task wird abnormal beendet.

Beispiel

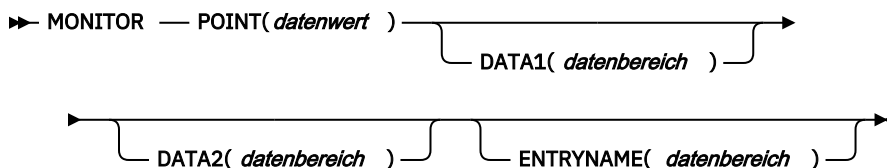
Das folgende Beispiel zeigt, wie eine benutzervorbereitete Tabelle mit dem Namen TB1 geladen wird:

```
EXEC CICS LOAD PROGRAM('TB1') SET(PTR)
```

MONITOR

Codiert einen Punkt zur Überwachung von Benutzerereignissen.

MONITOR



Bedingung: INVREQ

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl MONITOR stellt Informationen zur Leistung Ihrer Anwendungstransaktionen bereit. Er ersetzt die Überwachungsaspekte des Befehls ENTER TRACEID.

Neben den Überwachungsdaten, die an vordefinierten Ereignisüberwachungspunkten (EMPs - Event Monitoring Points) in CICS erfasst werden, kann ein Benutzeranwendungsprogramm Daten für Benutzerfelder in den CICS-Überwachungsdatensätzen beitragen. Zu diesem Zweck können Sie benutzerdefinierte EMPs mit dem Befehl MONITOR aufrufen. An jedem dieser EMPs können Sie 1 - 16384 Byte eigener Daten in jedem Leistungsüberwachungsdatensatz hinzufügen oder ändern. In diesen 16384 Byte können Sie eine Kombination der folgenden Elemente haben:

- 0 bis 256 Zähler
- 0 bis 256 Systemzeiten
- Eine einzelne 8192 Byte lange Zeichenfolge

Optionen

DATA1(datenbereich)

Gibt eine 4 Byte große Variable an, deren Inhalt vom Typ des verwendeten Benutzerereignisüberwachungspunkts abhängt:

- Wenn der Benutzerereignisüberwachungspunkt eine Option ADDCNT, SUBCNT, NACNT, EXCNT oder ORCNT enthält, ist die Variable DATA1 ein Datenbereich, wie er in der Überwachungssteuertabelle (MCT) der EMP-Definition für den Benutzer definiert ist.
- Wenn die Überwachungssteuertabelle (MCT) der EMP-Definition des Benutzers eine Option MLTCNT enthält, ist die Variable DATA1 ein Bereich mit der Adresse einer Serie von benachbarten Vollwörtern, die die Werte enthalten, die den Benutzerzählerfeldern hinzugefügt werden sollen, die in der Überwachungssteuertabelle (MCT) der EMP-Definition für den Benutzer definiert ist.
- Wenn die Überwachungssteuertabelle (MCT) der EMP-Definition des Benutzers eine Option MOVE enthält, ist die Variable DATA1 ein Bereich mit der Adresse der Zeichenfolge, die verschoben werden soll.

Details zu EMP-Benutzeroptionen finden Sie unter [Punkte der Benutzerereignisüberwachung: DFHMCT TYPE=EMP](#).

DATA2(datenbereich)

Gibt eine 4 Byte große Variable an, deren Inhalt vom Typ des verwendeten Benutzerereignisüberwachungspunkts abhängt:

- Wenn der Benutzerereignisüberwachungspunkt eine Option ADDCNT, SUBCNT, NACNT, EXCNT oder ORCNT enthält, ist die Variable DATA2 ein Datenbereich, wie er in der Überwachungssteuertabelle (MCT) der EMP-Definition für den Benutzer definiert ist.
- Wenn die Überwachungssteuertabelle (MCT) der EMP-Definition des Benutzers eine Option MLTCNT enthält, ist die Variable DATA2 ein Bereich mit der Anzahl der Benutzerzählerfelder, die aktualisiert werden sollen. Die Anzahl, die in DATA2 angegeben wird, überschreibt den Standardwert, der in der Überwachungssteuertabelle für die Operation definiert ist. Der Standardwert hängt von der Option ab, die Sie in der EMP-Definition definiert haben. Wenn Sie in DATA2 einen Nullwert angeben, verwendet die Überwachung den Standardwert, der in der EMP-Definition angegeben ist. Wenn DATA2 nicht angegeben wird, löst die MLTCNT-Operation eine Bedingung INVREQ aus, obwohl die Operation erfolgreich ausgeführt wurde.
- Wenn die Überwachungssteuertabelle (MCT) der EMP-Definition des Benutzers eine Option MOVE enthält, ist die Variable DATA2 ein Bereich mit der Länge der Zeichenfolge, die verschoben werden soll. Der Wert, der in DATA2 angegeben wird, überschreibt den Standardwert, der in der Überwachungssteuertabelle für die Operation definiert ist. Der Standardwert hängt von der Option ab, die Sie in der EMP-Definition definiert haben. Wenn Sie in DATA2 einen Nullwert angeben, verwendet die Überwachung den Standardwert, der in der EMP-Definition angegeben ist. Wenn DATA2 nicht angegeben wird, löst die MOVE-Operation eine Bedingung INVREQ aus, obwohl die Operation erfolgreich ausgeführt wurde.

In Ereignisüberwachungspunkte zur Anwendungsbenennung finden Sie ein Beispiel dafür, wie der Standardwert für einen Ereignisüberwachungspunkt (EMP) verarbeitet wird.

Details zu EMP-Benutzeroptionen finden Sie unter [Punkte der Benutzerereignisüberwachung: DFHMCT TYPE=EMP](#).

ENTRYNAME(datenbereich)

Gibt den Eintragsnamen des Überwachungspunkts an, der den Wert in POINT qualifiziert und in der Überwachungssteuertabelle (MCT) definiert ist. Als Wert für ENTRYNAME wird standardmäßig USER angenommen, wenn er nicht angegeben wird. Geben Sie in dem Datenbereich den Namen des acht Byte langen Felds in Ihrem Anwendungsprogramm an, das den Eintragsnamen des Überwachungspunkts enthält.

POINT(datenwert)

Gibt die Überwachungspunkt-ID wie in der Überwachungssteuertabelle (MCT) definiert an und liegt im Bereich von 0 - 255. Beachten Sie jedoch, dass Punkt-IDs im Bereich von 200 - 255 für die Verwendung durch IBM Programmprodukte reserviert sind.

Bedingungen

16 INVREQ

RESP2-Werte:

- 1** Der angegebene Wert für POINT liegt außerhalb des Bereichs von 1 bis 255.
- 2** Der angegebene Wert für POINT ist in der Überwachungssteuertabelle (MCT) nicht definiert.
- 3** Der angegebene Wert für DATA1 ist nicht gültig.
- 4** Der angegebene Wert für DATA2 ist nicht gültig.
- 5** Sie haben DATA1 für eine MCT-Operation nicht angegeben, die diese Angabe benötigt.
- 6** Sie haben DATA2 für eine MCT-Operation nicht angegeben, die diese Angabe benötigt.

Standardaktion: Die Task wird abnormal beendet.

Beispiele

Sie könnten diese Ereignisüberwachungspunkte (EMPs) für Benutzer zum Beispiel dazu verwenden, die Anzahl der Male zu zählen, die ein bestimmtes Ereignis auftritt, oder das Zeitintervall zwischen zwei Ereignissen zu messen.

Abbildung 2 auf Seite 342 enthält Beispiele für Befehle MONITOR (sowie der MCT-Einträge, die für die Beispiele erforderlich sind).

Anmerkung:

1. Beispiel 1 zeigt eine Benutzersystemuhr, die durch eine Anwendung gestartet wird, die als PROG3 identifiziert wird. Dies ist der 11. Ereignisüberwachungspunkt in dieser Anwendung. Um eine Verwechslung mit dem 11. Ereignisüberwachungspunkt in einer anderen Anwendung zu vermeiden, wird dieser Ereignisüberwachungspunkte durch den Tag ENTRY3.11 eindeutig identifiziert. Die Systemzeituhr, die gestartet wird, ist die erste Systemuhr in einer Zeichenfolge.
2. Beispiel 2 zeigt dieselbe Benutzersystemuhr, die von derselben Anwendung, jedoch durch einen anderen Ereignisüberwachungspunkt gestoppt wird. Der Ereignisüberwachungspunkt wird durch den Tag ENTRY3.12 eindeutig identifiziert.
3. Beispiel 3 zeigt einige Benutzerdaten, die in eine 32 Byte lange Zeichenfolge geladen werden, die für diesen Zweck reserviert ist. Das Laden beginnt an der relativen Position (Offset) 0 und die Daten sind nicht länger als 32 Byte.

```
1:
EXEC CICS MONITOR
    POINT(11)
    ENTRYNAME(ENTRY3)
    Erfordert: DFHMCT TYPE=EMP,
               CLASS=PERFORM,
               ID=(ENTRY3.11),
               CLOCK=(1,CLOCKA),
               PERFORM=SCLOCK(1)

2:
EXEC CICS MONITOR
    POINT(12)
    ENTRYNAME(ENTRY3)
    Erfordert: DFHMCT TYPE=EMP,
               CLASS=PERFORM,
               ID=(ENTRY3.12),
               PERFORM=PCLOCK(1)

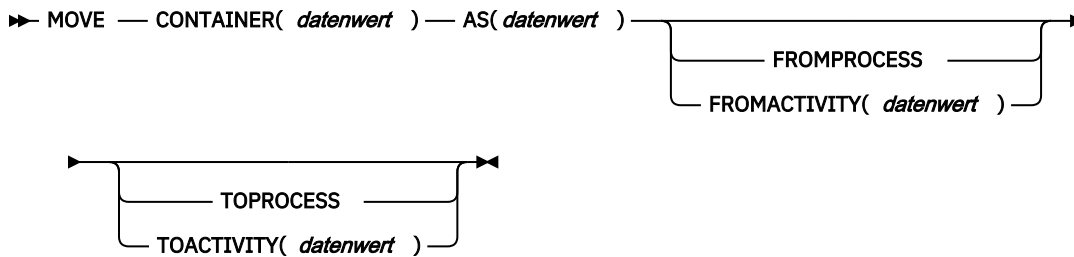
3:
EXEC CICS MONITOR
    POINT(13)
    DATA1(adresse_der_daten)
    DATA2(länge_der_daten)
    ENTRYNAME(ENTRY3)
    Erfordert: DFHMCT TYPE=EMP,
               CLASS=PERFORM,
               ID=(ENTRY3.13),
               PERFORM=MOVE(0,32)
```

Abbildung 2. Beispiele für die Codierung von Ereignisüberwachungspunkten (EMPs) für Benutzer

MOVE CONTAINER (BTS)

Verschiebt einen BTS-Datencontainer (und seinen Inhalt) aus einer Aktivität in eine andere.

MOVE CONTAINER (BTS)



Bedingungen: ACTIVITYERR, CONTAINERERR, INVREQ, IOERR, LOCKED

Beschreibung

Der Befehl `MOVE CONTAINER (BTS)` verschiebt einen Datencontainer (und seinen Inhalt) aus einer BTS-Aktivität in eine andere BTS-Aktivität. Nach dem Verschieben wird der Quellencontainer gelöscht.

Der Quellencontainer und der Zielcontainer werden durch den Namen und die Aktivitäten angegeben, die ihre Eigner sind. Die Aktivität, die Eigner des Quellencontainers ist, kann wie folgt angegeben werden:

- Explizit durch Angabe der Option `FROMPROCESS` oder `FROMACTIVITY`.
- Implizit durch Weglassen der Optionen `FROMPROCESS` und `FROMACTIVITY`. Wenn diese Optionen nicht angegeben werden, wird die aktuelle Aktivität impliziert.

Analog kann die Aktivität, die Eigner des Zielcontainers ist, wie folgt angegeben werden:

- Explizit durch Angabe der Option `TOPPROCESS` oder `TOACTIVITY`.
- Implizit durch Weglassen der Optionen `TOPPROCESS` und `TOACTIVITY`. Wenn diese Optionen nicht angegeben werden, wird die aktuelle Aktivität impliziert.

Sie können einen Container wie folgt verschieben:

- Aus der aktuellen Aktivität in eine untergeordnete Aktivität der aktuellen Aktivität.
- Aus einer untergeordneten Aktivität der aktuellen Aktivität in die aktuelle Aktivität.
- Aus einer aktuellen Aktivität in die aktuelle Aktivität (Umbenennen des Containers).
- Aus einer untergeordneten Aktivität der aktuellen Aktivität in eine andere untergeordnete Aktivität.

Darüber hinaus können Sie einen Container wie folgt verschieben, *wenn die aktuelle Aktivität die Stammaktivität ist*:

- Aus dem aktuellen Prozess in die aktuelle Aktivität (Stammaktivität).
- Aus dem aktuellen Prozess in eine untergeordnete Aktivität der aktuellen Aktivität.
- Aus dem aktuellen Prozess in den aktuellen Prozess (Umbenennen des Containers).
- Aus der aktuellen Aktivität in den aktuellen Prozess.
- Aus einer untergeordneten Aktivität der aktuellen Aktivität in den aktuellen Prozess.

Sie können den Befehl `MOVE CONTAINER` anstelle der Befehle `GET CONTAINER` und `PUT CONTAINER` als effizientere Methode zur Übertragung von Daten zwischen Aktivitäten verwenden. Eine Erläuterung finden Sie unter [Containerbefehle](#).

Anmerkung:

1. Wenn der Quellencontainer nicht vorhanden ist, tritt ein Fehler auf.
2. Wenn der Zielcontainer noch nicht vorhanden ist, wird er erstellt. Ist der Zielcontainer bereits vorhanden, wird der vorherige Inhalt überschrieben.
3. Sie können Container nicht aus einem Prozess in einen anderen Prozess verschieben. Sowohl der Quellencontainer als auch der Zielcontainer müssen sich im Bereich des aktuellen Prozesses befinden.
4. Nur die Stammaktivität kann einen Prozesscontainer als Quelle oder Ziel für einen Befehl `MOVE CONTAINER` angeben.

Die Container eines Prozesses sind *nicht* mit den Containern der zugehörigen Stammaktivität identisch.

Sie auch „GET CONTAINER (BTS)“ auf Seite 233 und „PUT CONTAINER (BTS)“ auf Seite 353.

Optionen

AS(datenwert)

Gibt den Namen (1 - 16 Zeichen) des Zielcontainers an. Ist der Zielcontainer bereits vorhanden, wird der Inhalt überschrieben.

CONTAINER(datenwert)

Gibt den Namen (1 - 16 Zeichen) des Quellencontainers an, der verschoben werden soll.

FROMACTIVITY(datenwert)

Gibt den Namen (1 - 16 Zeichen) der Aktivität an, die Eigner des Quellencontainers ist. Wenn angegeben, muss diese Option eine untergeordnete Aktivität der aktuellen Aktivität (oder die aktuelle Aktivität selbst) angeben.

FROMPROCESS

Gibt an, dass der Quellencontainer dem aktuellen Prozess gehört, das heißt, dem Prozess, für den das Programm ausgeführt wird, das den Befehl absetzt.

TOACTIVITY(datenwert)

Gibt den Namen (1 - 16 Zeichen) der Aktivität an, die Eigner des Zielcontainers ist. Wenn angegeben, muss diese Option eine untergeordnete Aktivität der aktuellen Aktivität (oder die aktuelle Aktivität selbst) angeben.

TOPROCESS

Gibt an, dass der Zielcontainer dem aktuellen Prozess gehört, das heißt, dem Prozess, für den das Programm ausgeführt wird, das den Befehl absetzt.

Bedingungen

109 ACTIVITYERR

RESP2-Werte:

8

Die in der Option FROMACTIVITY oder TOACTIVITY angegebene Aktivität konnte nicht gefunden werden.

110 CONTAINERERR

RESP2-Werte:

10

Der in der Option CONTAINER angegebene Container konnte nicht gefunden werden.

26

Der in der Option CONTAINER angegebene Prozesscontainer ist schreibgeschützt.

16 INVREQ

RESP2-Werte:

4

Der Befehl wurde außerhalb des Gültigkeitsbereichs einer momentan aktiven Aktivität abgesetzt.

25

Die Option FROMPROCESS oder TOPROCESS wurde verwendet, jedoch wurde der Befehl außerhalb des Gültigkeitsbereichs eines momentan aktiven Prozesses abgesetzt.

17 IOERR

RESP2-Werte:

30

Es ist ein Ein-/Ausgabefehler für die Repository-Datei aufgetreten.

31

Der Datensatz in der Repository-Datei wird gerade verwendet.

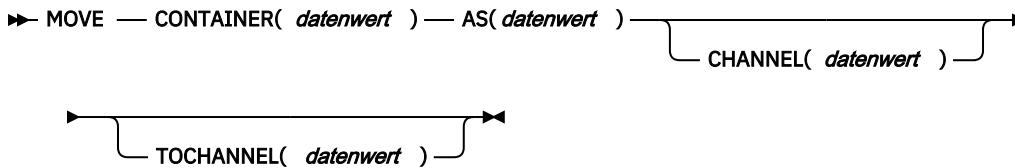
100 LOCKED

Die Anforderung kann nicht ausgeführt werden, weil eine Sperre für den relevanten Datensatz in der Repository-Datei besteht.

MOVE CONTAINER (CHANNEL)

Verschiebt einen Container (und seinen Inhalt) aus einem Kanal in einen anderen.

MOVE CONTAINER (CHANNEL)



Bedingungen: CHANNELERR, CONTAINERERR, INVREQ

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl `MOVE CONTAINER (CHANNEL)` verschiebt einen Container aus einem Kanal in einen anderen. Nach dem Verschieben ist der Quellencontainer nicht mehr vorhanden.

Der Quellencontainer und der Zielcontainer werden durch den Namen und die Kanäle angegeben, die ihre Eigner sind. Der Kanal, der Eigner des Quellencontainers ist, kann wie folgt angegeben werden:

- Explizit durch Angabe der Option `CHANNEL`.
- Implizit durch Weglassen der Option `CHANNEL`. Wenn diese Option nicht angegeben wird, wird der aktuelle Kanal impliziert.

Ganz ähnlich kann der Kanal, der Eigner des Zielcontainers ist, wie folgt angegeben werden:

- Explizit durch Angabe der Option `TOCHANNEL`.
- Implizit durch Weglassen der Option `TOCHANNEL`. Wenn diese Option nicht angegeben wird, wird der aktuelle Kanal impliziert.

Sie können einen Container wie folgt verschieben:

- Aus einem Kanal in einen anderen.
- Innerhalb desselben Kanals - zum Beispiel aus dem aktuellen Kanal in den aktuellen Kanal. Dies hat die Wirkung einer Umbenennung des Containers.

Sie können den Befehl **MOVE CONTAINER** anstelle der Befehle **GET CONTAINER** und **PUT CONTAINER** als effizientere Methode zur Übertragung von Daten zwischen Kanälen verwenden.

Anmerkung:

1. Der Quellenkanal muss sich im Bereich des Programms befinden, das den Befehl **MOVE CONTAINER** absetzt.
2. Wenn der Zielkanal im Bereich des Programms, das den Befehl **MOVE CONTAINER** absetzt, nicht vorhanden ist, wird er erstellt.
3. Wenn der Quellencontainer nicht vorhanden ist, tritt ein Fehler auf.
4. Wenn der Zielcontainer noch nicht vorhanden ist, wird er erstellt. Ist der Zielcontainer bereits vorhanden, wird der vorherige Inhalt überschrieben.
5. Wenn Sie einen Container mit sich selbst überschreiben, findet keine Operation statt. Das heißt, wenn Sie für die Optionen `CONTAINER` und `AS` dieselben Werte angeben und entweder beide Optionen `CHANNEL` und `TOCHANNEL` nicht angeben oder sie mit denselben Werten angeben, sodass derselbe Kanal angegeben wird, wird der Quellencontainer nicht geändert und nicht gelöscht. Es wird keine Fehlerbedingung ausgelöst.

Optionen

AS(datenwert)

Gibt den Namen (1 - 16 Zeichen) des Zielcontainers an. Ist der Zielcontainer bereits vorhanden, wird der Inhalt überschrieben.

Zulässige Zeichen: A-Z a-z 0-9 \$ @ # / % & ? ! : | " = ~ , ; < > . - und _ . Führende und eingebettete Leerzeichen sind nicht zulässig. Wenn der angegebene Name kürzer als 16 Zeichen ist, wird er mit nachfolgenden Leerzeichen bis auf 16 Zeichen aufgefüllt.

Containernamen haben immer EBCDIC-Format. Der zulässige Satz an Zeichen für Containernamen, wie oben aufgeführt, umfasst einige Zeichen, die nicht in allen EBCDIC-Codepages dieselbe Darstellung haben. Daher wird empfohlen, wenn Kanäle zwischen Regionen übertragen werden sollen, die Zeichen zur Kanalbenennung auf die folgenden zu begrenzen: A-Z a-z 0-9 & : = , ; < > . - und _ .

CHANNEL(datenwert)

Gibt den Namen (1 - 16 Zeichen) des Kanals an, der Eigner des Quellencontainers ist. Sie können den Kanalnamen DFHTRANSACTION angeben, um den Transaktionskanal zu verwenden. Wenn diese Option nicht angegeben wird, wird der aktuelle Kanal impliziert.

CONTAINER(datenwert)

Gibt den Namen (1 - 16 Zeichen) des Quellencontainers an, der verschoben werden soll.

TOCHANNEL(datenwert)

Gibt den Namen (1 - 16 Zeichen) des Kanals an, der Eigner des Zielcontainers ist. Zulässige Zeichen: A-Z a-z 0-9 \$ @ # / % & ? ! : | " = , ; > . - und _ . Führende und eingebettete Leerzeichen sind nicht zulässig. Wenn der angegebene Name kürzer als 16 Zeichen ist, wird er mit nachfolgenden Leerzeichen bis auf 16 Zeichen aufgefüllt. Wenn der Kanal nicht vorhanden ist, wird er erstellt. Dieser neue Kanal bleibt im Bereich, bis die Verbindungsebene geändert wird. Weitere Informationen zum Kanalbereich finden Sie unter [Geltungsbereich eines Kanals](#).

Kanalnamen haben immer EBCDIC-Format. Der Satz an zulässigen Zeichen für Kanalnamen, wie zuvor aufgeführt, umfasst einige Zeichen, die nicht in allen EBCDIC-Codepages dieselbe Darstellung haben. Wenn Kanäle zwischen Regionen übertragen werden sollen, wird daher empfohlen, die Zeichen zur Kanalbenennung auf die folgenden zu begrenzen: A-Z a-z 0-9 : = , ; > . - und _ .

Sie können den Kanalnamen DFHTRANSACTION angeben, um einen Transaktionskanal zu verwenden. Ein Transaktionskanal verlässt den Bereich nicht, wenn die Verbindungsebene geändert wird: er ist in der Transaktion immer zugänglich. Weitere Informationen finden Sie unter [Kanäle und Container](#).

Wenn diese Option nicht angegeben wird, wird der aktuelle Kanal impliziert.

Bedingungen

122 CHANNELERR

RESP2-Werte:

1

Der Name in der Option TOCHANNEL enthält ein ungültiges Zeichen oder eine ungültige Kombination von Zeichen.

2

Der in der Option CHANNEL angegebene Kanal konnte nicht gefunden werden.

3

Entweder der aktuelle Kanal oder der in der Option CHANNEL angegebene Kanal ist schreibgeschützt.

110 CONTAINERERR

RESP2-Werte:

10

Der in der Option CONTAINER angegebene Container konnte nicht gefunden werden.

18

Der Name in der Option AS enthält ein ungültiges Zeichen oder eine ungültige Kombination von Zeichen.

16 INVREQ

RESP2-Werte:

4

Die Option CHANNEL oder TOCHANNEL wurde nicht angegeben, es ist kein aktueller Kanal vorhanden (weil an das Programm, das den Befehl abgesetzt hat, kein Kanal übergeben wurde) und der Befehl wurde außerhalb des Gültigkeitsbereichs der momentan aktiven BTS-Aktivität abgesetzt.

30

Sie können keinen CICS-definierten schreibgeschützten Container verschieben.

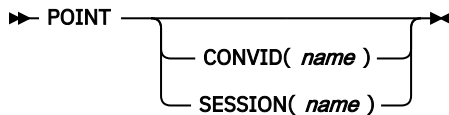
31

Sie können keinen Container in einen vorhandenen CICS-definierten und schreibgeschützten Container verschieben (d. h. diesen überschreiben).

POINT

Ruft Informationen zu einer logischen Einheit vom LU-Typ 6.1 (LUTYPE6.1) ab.

POINT



Bedingung: NOTALLOC

Beschreibung

Der Befehl POINT ruft Informationen zu einer benannten Funktion ab, wie zum Beispiel, ob die Sitzung Eigner der bestimmten Funktion ist.

Dieser Befehl kann für eine MRO-Sitzung verwendet werden.

Optionen

CONVID(name)

Gibt den Dialog an, auf den sich der Befehl bezieht. Der vier Zeichen lange Name bezeichnet entweder das Token, das von einem zuvor ausgeführten Befehl ALLOCATE in EIBRSRCE im EIB zurückgegeben wurde, oder das Token, das die Hauptsitzung (die durch einen zuvor ausgeführten Befehl ASSIGN zurückgegeben wurde) darstellt.

SESSION(name)

Gibt die symbolische ID (1 - 4 Zeichen) für einen Terminaleintrag in der Terminalsteuertabelle (TCTTE) für die Sitzung an. Diese Option gibt die alternative Funktion an, die verwendet werden soll. Wenn weder diese Option noch die Option CONVID angegeben werden, wird die Hauptfunktion für die Task verwendet.

Bedingungen

61 NOTALLOC

Tritt auf, falls die Anwendung nicht Eigner der im Befehl angegebenen Funktion ist.

Standardaktion: Die Task wird abnormal beendet.

POP HANDLE

Stellt den Stack wieder her.

POP HANDLE

➡ POP HANDLE ⬅

Bedingung: INVREQ

Dieser Befehl ist threadsicher.

Beschreibung

Verwenden Sie den Befehl **POP HANDLE**, um die Wirkung der Befehle **IGNORE CONDITION**, **HANDLE ABEND**, **HANDLE AID** und **HANDLE CONDITION** in dem Status wiederherzustellen, in dem sie sich befanden, bevor ein Befehl **PUSH HANDLE** auf der aktuellen Verbindungsebene ausgeführt wurde.

Einschränkung: Dieser Befehl wird nur in Anwendungen unterstützt, die in COBOL, PL/I und Assemblersprache geschrieben sind (jedoch nicht in Anwendungen in Assemblersprache mit AMODE(64)). In allen anderen unterstützten höheren Programmiersprachen wird der Befehl nicht unterstützt.

Dieser Befehl kann zum Beispiel während einer Verzweigung zu einer Subroutine nützlich sein, die in ein Hauptprogramm eingebettet ist.

Wenn ein CICS-Programm eine Subroutine (auf derselben logischen Ebene) aufruft, übernimmt das Programm oder die Routine, die die Steuerung empfängt, normalerweise die aktuellen Befehle für **HANDLE**. Diese Befehle sind in dem aufgerufenen Programm möglicherweise nicht geeignet. Das aufgerufene Programm kann den Befehl **PUSH HANDLE** verwenden, um vorhandene Befehle **HANDLE** auszusetzen. Vor der Rückgabe der Steuerung an das aufrufende Programm können die ursprünglichen Befehle mit dem Befehl **POP HANDLE** wiederhergestellt werden.

Anmerkung: Wenn ein CICS-Programm den Befehl EXEC CICS LINK verwendet, um ein anderes CICS-Programm aufzurufen, werden die HANDLE-Wirkungen nicht von dem verknüpften Programm übernommen, aber CICS sucht auf den vorangehenden logischen Ebenen nach einem Exit **HANDLE ABEND**. Weitere Informationen zur Beziehung zwischen LINK und HANDLE ABEND finden Sie unter [Wiederherstellung nach abnormaler Beendigung](#).

Sie können Folgen des Befehls **PUSH HANDLE ... POP HANDLE** innerhalb einer Task verschachteln. Jeder Befehl POP HANDLE stellt einen Satz von Spezifikationen wieder her.

Bedingungen

16 INVREQ

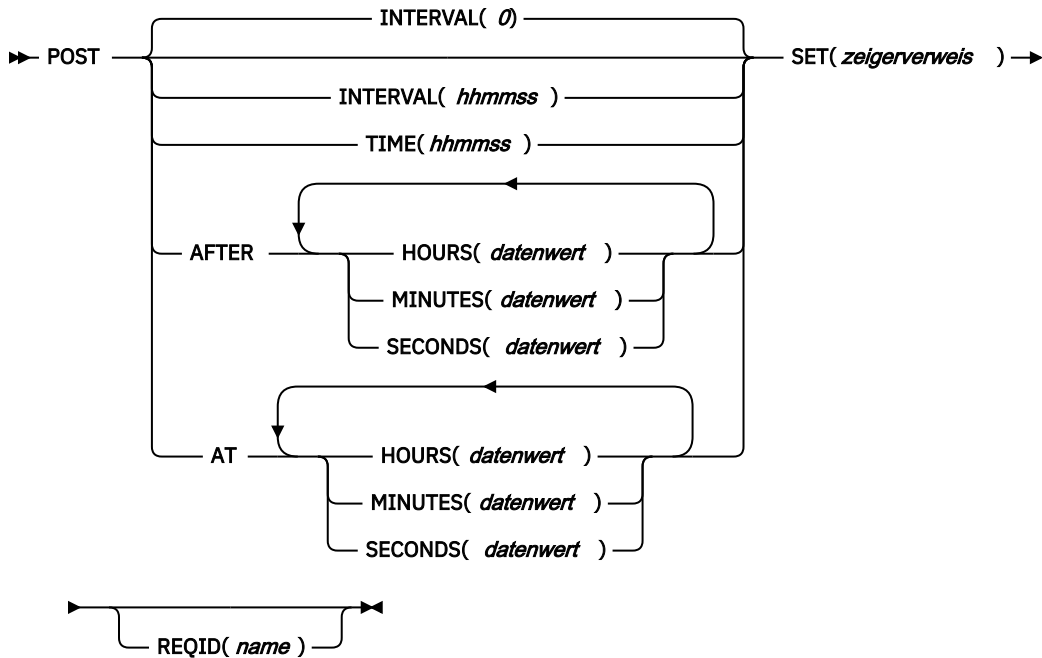
Tritt auf, wenn kein entsprechender Befehl **PUSH HANDLE** auf der aktuellen Verbindungsebene ausgeführt wurde.

Standardaktion: Die Task wird abnormal beendet.

POST

Fordert eine Benachrichtigung an, wenn eine angegebene Zeit abgelaufen ist.

POST



Bedingungen: EXPIRED, INVREQ

Hinweis für dynamisches Transaktionsrouting: Die Verwendung von POST kann bei späterem Abbruch durch CANCEL durch eine andere Task möglicherweise transaktionsübergreifende Affinitäten erstellen, die sich negativ auf die Verwendung des dynamischen Transaktionsroutings auswirken. Weitere Informationen zu Transaktionsaffinitäten finden Sie unter [Affinität](#).

Beschreibung

Der Befehl POST fordert eine Benachrichtigung darüber an, dass eine angegebene Zeit abgelaufen ist. Als Antwort auf diesen Befehl macht CICS einen Steuerbereich für ein Zeitgeberereignis verfügbar. Dieser vier Byte lange Steuerbereich wird mit binären Nullen initialisiert und der Zeigerverweis, der in der Option SET angegeben wird, wird auf die Adresse des Steuerbereichs gesetzt.

Wenn die Zeit, die Sie angeben, abgelaufen ist, wird der Steuerbereich für das Zeitgeberereignis gepostet. Das heißt, das erste Byte wird auf den Wert X'40' und das dritte Byte auf den Wert X'80' gesetzt. Sie können das Posten auf eine der folgenden Arten testen:

- Durch Prüfen des Steuerbereichs für das Zeitgeberereignis in Intervallen. Sie müssen CICS die Möglichkeit geben, den Bereich zu posten. Das heißt, die Task muss die Steuerung von CICS abgeben, bevor Sie den Bereich testen. Normalerweise wird diese Bedingung durch andere Befehle, die abgesetzt werden, erfüllt. Wenn eine Task eine lange interne Funktion ausführt, können Sie erzwingen, dass die Steuerung abgegeben wird, indem Sie einen Befehl SUSPEND absetzen.
- Durch Aussetzen der Taskaktivität durch einen Befehl WAIT EVENT oder WAIT EXTERNAL, bis der Steuerbereich des Zeitgeberereignisses gepostet wird. Diese Aktion ist dem Absetzen eines Befehls DELAY ähnlich, jedoch können Sie mit der Befehlsabfolge POST und WAIT EVENT oder WAIT EXTERNAL eine Verarbeitung nach dem Befehl POST ausführen. Ein Befehl DELAY setzt die Taskaktivität sofort aus. Keine andere Task sollte versuchen, auf das Ereignis zu warten, das durch einen Befehl POST eingerichtet wird.
- Durch Verwenden des Befehls WAITCICS.

Der Steuerbereich für das Zeitgeberereignis, der einem Befehl POST zugeordnet ist, ist vorhanden, wenn die Transaktion, die den Befehl POST abgesetzt hat, noch besteht und keinen der folgenden Befehle abgesetzt hat:

- Nachfolgende Befehle POST

- Befehl CANCEL für den Befehl POST
- Lokaler Befehl START

Ein Befehl START, der eine Transaktion auf einem fernen System benennt, hat keine Auswirkung auf das Ereignis, das durch den POST eingerichtet wurde, sofern die Transaktion nicht mit dem Wert YES für den Parameter LOCALQ auf YES definiert ist und eine lokale Warteschlangensteuerung stattfindet.

- Nachfolgende Befehle DELAY

Der Steuerbereich für das Zeitgeberereignis wird gepostet, wenn die Transaktion, die den Befehl POST abgesetzt hat, vorhanden ist und eines der folgenden Ereignisse auftritt:

- Die durch den Befehl POST angegebene Zeit läuft ab.
- Eine andere Transaktion setzt einen Befehl CANCEL mit der REQID des Befehls POST ab.

Der Steuerbereich für das Zeitgeberereignis kann aus mehreren Gründen freigegeben werden. Wenn dies geschieht, ist das Ergebnis eines Befehls WAIT, der von einer beliebigen anderen Task für das Ereignis abgesetzt wird, das durch den Befehl POST eingerichtet wurde, unvorhersehbar.

Allerdings können andere Tasks das Ereignis abbrechen, wenn sie Zugriff auf die Anforderungs-ID (REQID) haben, die dem Befehl POST zugeordnet ist. (Siehe Befehl „CANCEL“ auf Seite 67 und die Beschreibung der Option REQID.)

Eine Task kann nur einen aktiven Befehl POST zu gleicher Zeit haben. Ein Befehl DELAY oder POST oder aber ein Befehl START, der eine Transaktion im lokalen System benennt, ersetzt einen Befehl POST, der zuvor von der Task abgesetzt wurde.

Das Standardintervall ist INTERVAL(0), in der Sprache C ist die Standardangabe jedoch AFTER HOURS(0) MINUTES(0) SECONDS(0).

Optionen

AFTER

Gibt das Intervall der Zeit an, die ablaufen soll.

Es gibt zwei Möglichkeiten, die Zeit unter AFTER und AT einzugeben.

1. Als Kombination von mindestens zwei der Optionen HOURS(0-99), MINUTES(0-59) und SECONDS(0-59). Zum Beispiel würde die Angabe HOURS(1) SECONDS(3) eine Stunde und drei Sekunden bedeuten (für die Minuten wird standardmäßig null angenommen).
2. Durch einen der Werte HOURS(0-99), MINUTES(0-5999) oder SECONDS(0-359 999). Die Angabe HOURS(1) bedeutet 1 Stunde. MINUTES(62) bedeutet eine Stunde und zwei Minuten. Die Angabe SECONDS(3723) bedeutet 1 Stunde, 2 Minuten und 3 Sekunden.

AT

Gibt den Zeitpunkt des Ablaufs an. Informationen zu den Möglichkeiten der Zeitangabe finden Sie unter der Option AFTER.

HOURS(datenwert)

Gibt einen Vollwort-Binärwert im Bereich von 0 bis 99 an. Dies ist eine Unteroption der Optionen AFTER und AT. Informationen zur Verwendung und zur Bedeutung finden Sie unter der Option AFTER.

INTERVAL(hhmmss)

Gibt ein Zeitintervall an, das ab dem Zeitpunkt, zu dem der Befehl POST abgesetzt wird, ablaufen soll. Die Werte **hh** und **ss** liegen im Bereich von 0 bis 59. Die angegebene Zeit wird von CICS der aktuellen Systemzeit hinzuaddiert, wenn der Befehl ausgeführt wird, um die Ablaufzeit zu berechnen.

Diese Option wird verwendet, um anzugeben, wann das Posten des Steuerbereichs für das Zeitgeberereignis stattfinden soll.

Bei Verwendung der Programmiersprache C wird empfohlen, die Optionen AFTER/AT HOURS, MINUTES und SECONDS zu verwenden, da C keinen gepackten Dezimaldatentyp bereitstellt. Sie können INTERVAL verwenden, wenn jedoch der angegebene Wert **keine** ganzzahlige Konstante ist, obliegt es der Anwendung, sicherzustellen, dass der an CICS übergebene Wert das gepackte Dezimalformat hat.

MINUTES(datenwert)

Gibt einen Vollwort-Binärwert aus dem Bereich von 0 bis 59 an, wenn auch die Optionen HOURS oder SECONDS angegeben werden, oder gibt einen Wert aus dem Bereich von 0 bis 5999 an, wenn MINUTES die einzige Option ist, die angegeben wird. Dies ist eine Unteroption der Optionen AFTER und AT. Informationen zur Verwendung und zur Bedeutung finden Sie unter der Option AFTER.

REQID(name)

Gibt einen Namen (1-8 Zeichen) an, der eindeutig sein muss, um die POST-Anforderung zu identifizieren. Die Verwendung dieser Option zur Angabe eines anwendungsdefinierten Namens ist die einzige Methode, einer anderen Transaktion die Möglichkeit zu geben, die POST-Anforderung abzubrechen (CANCEL).

Wenn Sie keine eigene Anforderungs-ID angeben, generiert CICS eine eindeutige Anforderungs-ID für Sie im Feld EIBREQID des EXEC-Schnittstellenblocks. Diese kann wie eine eigene Anforderungs-ID (REQID) von einer anderen Transaktion zum Abbrechen der POST-Anforderung verwendet werden.

Wenn andere Tasks noch nicht abgelaufene POST-Anforderungen abbrechen können sollen, müssen Sie die Anforderungs-ID dynamisch verfügbar machen. Das Speichern einer Anforderungs-ID in einer TS-Warteschlange, deren Name anderen Anwendungen bekannt ist, die die POST-Anforderung möglicherweise abbrechen sollen, ist eine Methode, mit der Sie eine Anforderungs-ID an andere Transaktionen übergeben können.

SECONDS(datenwert)

Gibt einen Vollwort-Binärwert aus dem Bereich von 0 bis 59 an, wenn auch die Optionen HOURS oder MINUTES angegeben werden, oder gibt einen Wert aus dem Bereich von 0 bis 359 999, wenn SECONDS die einzige Option ist, die angegeben wird. Dies ist eine Unteroption der Optionen AFTER und AT. Informationen zur Verwendung und zur Bedeutung finden Sie unter der Option AFTER.

SET(zeigerverweis)

Gibt den Zeigerverweis an, der auf die Adresse des vier Byte großen Steuerbereichs für das Zeitgeberereignis gesetzt werden soll, der von CICS generiert wird. Dieser Bereich wird mit binären Nullen initialisiert. Nach Ablauf der angegebenen Zeit wird das Byte auf den Wert X'40' und das dritte Byte auf den Wert X'80' gesetzt.

Der Steuerbereich für das Zeitgeberereignis befindet sich immer unterhalb der 16-MB-Grenze im gemeinsam genutzten dynamischen Speicher (SDSA).

TIME(hhmmss)

Gibt den Zeitpunkt an, zu dem das Posten des Steuerbereichs für das Zeitgeberereignis stattfinden soll.

Bei Verwendung der Programmiersprache C wird empfohlen, die Optionen AFTER/AT HOURS, MINUTES und SECONDS zu verwenden, da C keinen gepackten Dezimaldatentyp bereitstellt. Sie können TIME verwenden, wenn jedoch der angegebene Wert **keine** ganzzahlige Konstante ist, obliegt es der Anwendung, sicherzustellen, dass der an CICS übergebene Wert das gepackte Dezimalformat hat. Informationen zu Ablaufzeiten finden Sie unter [Intervallsteuerung](#).

Bedingungen**31 EXPIRED**

Tritt auf, wenn die angegebene Zeit bereits abgelaufen ist, wenn der Befehl abgesetzt wird.

Standardaktion: Die Bedingung wird ignoriert.

16 INVREQ

RESP2-Werte:

- 4** Stundenangabe außerhalb des gültigen Bereichs.
- 5** Minutenangabe außerhalb des gültigen Bereichs.
- 6** Sekundenangabe außerhalb des gültigen Bereichs.

Tritt außerdem in jedem der folgenden Fälle auf (RESP2 nicht festgelegt):

- Der Befehl POST ist für die Verarbeitung durch CICS nicht gültig.

Standardaktion: Die Task wird abnormal beendet.

Beispiele

Das folgende Beispiel zeigt, wie ein Steuerbereich für ein Zeitgeberereignis für eine Task angefordert wird, der nach 30 Sekunden gepostet werden soll:

```
EXEC CICS POST  
      INTERVAL(30)  
      REQID('RBL3D')  
      SET(PREF)
```

Das folgende Beispiel zeigt, wie eine Benachrichtigung angefordert wird, wenn die angegebene Tageszeit erreicht wird. Da in dem Befehl keine Anforderungs-ID angegeben wird, weist CICS automatisch eine Anforderungs-ID zu und gibt sie an das Anwendungsprogramm im Feld EIBREQID im EIB zurück.

```
EXEC CICS POST  
      TIME(PACKTIME)  
      SET(PREF)
```

PURGE MESSAGE

Stellt die Erstellung einer logischen BMS-Nachricht ein.

PURGE MESSAGE

➡ PURGE MESSAGE ➡

Bedingungen: Voller BMS: INVREQ, TSIOERR

Beschreibung

Der Befehl PURGE MESSAGE beendet die Erstellung einer logischen BMS-Nachricht. Er löscht die aktuelle logische Nachricht, einschließlich aller Seiten eines einheitenabhängigen Datenstroms, der bereits an den temporären CICS-Speicher geschrieben wurde. Das Anwendungsprogramm kann anschließend eine neue logische Nachricht erstellen.

Die Teile der logischen Nachricht, die bereits im Hauptspeicher oder im temporären Speicher erstellt wurden, werden gelöscht.

Informationen zu Maskendefinitionsmakros finden Sie unter [BMS-Makros](#).

Der Befehl PURGE MESSAGE ist nur in BMS mit vollem Funktionsumfang verfügbar. Weitere Informationen zu BMS finden Sie unter [Basic Mapping Support](#).

Bedingungen

16 INVREQ

RESP2-Werte:

200

Der Befehl wurde in einem Distributed Program Link-Serverprogramm aufgerufen.

Standardaktion: Die Task wird abnormal beendet.

35 TSIOERR

Tritt auf, falls ein nicht behebbarer Ein-/Ausgabefehler für den temporären Speicher aufgetreten ist.

Standardaktion: Die Task wird abnormal beendet.

PUSH HANDLE

Setzt den Stack aus.

PUSH HANDLE

➡ PUSH HANDLE ➡

Dieser Befehl ist threadsicher.

Beschreibung

Mit dem Befehl **PUSH HANDLE** können Sie die aktuelle Wirkung von Befehlen **IGNORE CONDITION**, **HANDLE ABEND**, **HANDLE AID** und **HANDLE CONDITION** aussetzen.

Einschränkung: Dieser Befehl wird nur in Anwendungen unterstützt, die in COBOL, PL/I und Assemblersprache geschrieben sind (jedoch nicht in Anwendungen in Assemblersprache mit AMODE(64)). In allen anderen unterstützten höheren Programmiersprachen wird der Befehl nicht unterstützt.

Dieser Befehl kann zum Beispiel während einer Verzweigung zu einer Subroutine nützlich sein, die in ein Hauptprogramm eingebettet ist.

Wenn ein CICS-Programm eine Subroutine auf derselben logischen Ebene aufruft, übernimmt das Programm oder die Routine, die die Steuerung empfängt, normalerweise die aktuellen Befehle für **HANDLE**. Diese Befehle sind in dem aufgerufenen Programm möglicherweise nicht geeignet. Das aufgerufene Programm kann den Befehl **PUSH HANDLE** verwenden, um vorhandene Befehle für **HANDLE** auszusetzen.

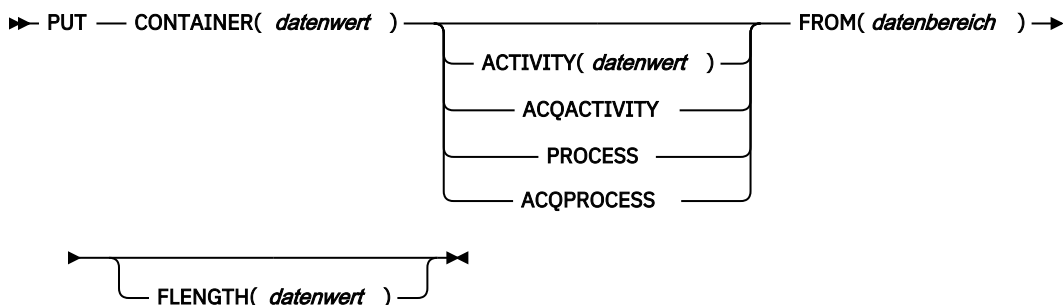
Anmerkung: Wenn ein CICS-Programm den Befehl EXEC CICS LINK verwendet, um ein anderes CICS-Programm aufzurufen, werden die Optionen von **HANDLE CONDITION** nicht von dem verknüpften Programm übernommen, aber CICS sucht auf den vorangehenden logischen Ebenen nach einem Exit **HANDLE ABEND**. Weitere Informationen zur Beziehung zwischen LINK und HANDLE ABEND finden Sie unter [Wiederherstellung nach abnormaler Beendigung](#).

Sie können Folgen des Befehls **PUSH HANDLE ... POP HANDLE** innerhalb einer Task verschachteln. Jeder Befehl **PUSH HANDLE** verschiebt einen Satz von Spezifikationen in den Stack.

PUT CONTAINER (BTS)

Speichert Daten in einem angegebenen BTS-Datencontainer.

PUT CONTAINER (BTS)



Bedingungen: ACTIVITYERR, CONTAINERERR, INVREQ, IOERR, LOCKED, PROCESSBUSY

Beschreibung

Der Befehl PUT CONTAINER (BTS) speichert Daten und platziert sie in einem Container, der einer angegebenen BTS-Aktivität oder einem angegebenen BTS-Prozess zugeordnet ist.

Der Container wird durch den Namen angegeben. Der Prozess oder die Aktivität, der bzw. die Eigner des Containers ist, kann wie folgt angegeben werden:

- Explizit durch Angabe einer der Optionen für Prozess oder Aktivität.
- Implizit durch Weglassen der Optionen für Prozess oder Aktivität. Wenn diese Optionen nicht angegeben werden, wird die aktuelle Aktivität impliziert.

Anmerkung:

1. Es gibt keine Begrenzung für die Anzahl von Containern, die einer Aktivität zugeordnet werden können.
2. Verschiedene Aktivitäten können Eigner identisch benannter Container sein. Diese Container sind unterschiedliche Container.
3. Wenn der angegebene Container nicht bereits vorhanden ist, wird er erstellt. Ist der angegebene Container bereits vorhanden, wird der vorherige Inhalt überschrieben.
4. Container, deren Eigner ein Prozess ist (*Prozesscontainer*) können von jeder Aktivität in dem Prozess gelesen werden. Sie können jedoch nur von der Stammaktivität oder durch ein Programm, das den Prozess angefordert hat, aktualisiert werden.

Die Container eines Prozesses sind *nicht* mit den Containern der zugehörigen Stammaktivität identisch.

Sie auch „[GET CONTAINER \(BTS\)](#)“ auf Seite 233 und „[MOVE CONTAINER \(BTS\)](#)“ auf Seite 342.

Optionen

ACQACTIVITY

Gibt eine der folgenden Informationen an:

- Wenn das Programm, das den Befehl absetzt, einen Prozess angefordert hat, gibt diese Option an, dass der Eigner des Containers die Stammaktivität dieses Prozesses ist.
- Andernfalls gibt diese Option an, dass der Container der Aktivität gehört, die von dem Programm durch einen Befehl ACQUIRE ACTIVITYID angefordert wurde.

ACQPROCESS

Gibt an, dass der Container dem Prozess gehört, den das Programm, das den Befehl absetzt, in der aktuellen Arbeitseinheit angefordert hat.

ACTIVITY(datenwert)

Gibt den Namen (1-16 Zeichen) der Aktivität an, die Eigner des Containers ist. Dies muss eine untergeordnete Aktivität der aktuellen Aktivität sein.

CONTAINER(datenwert)

Gibt den Namen (1-16 Zeichen) des Containers an, in den die Daten eingefügt werden sollen.

Zulässige Zeichen: A-Z a-z 0-9 \$ @ # / % & ? ! : | " = ~ , ; < > . - und _ . Führende und eingebettete Leerzeichen sind nicht zulässig. Wenn der angegebene Name kürzer als 16 Zeichen ist, wird er mit nachfolgenden Leerzeichen bis auf 16 Zeichen aufgefüllt.

FLENGTH(datenwert)

Gibt die Länge des Datenbereichs, aus dem Daten gelesen werden sollen, als Vollwort-Binärwert an.

FROM(datenbereich)

Gibt einen Bereich des Arbeitsspeichers an, aus dem die zu speichernden Daten gelesen werden sollen.

PROCESS

Gibt an, dass der Container, in dem Daten platziert werden sollen, dem aktuellen Prozess gehört, das heißt, dem Prozess, für den das Programm ausgeführt wird, das den Befehl absetzt.

Bedingungen

109 ACTIVITYERR

RESP2-Werte:

8

Die in der Option ACTIVITY angegebene Aktivität konnte nicht gefunden werden.

110 CONTAINERERR

RESP2-Werte:

10

Der in der Option CONTAINER angegebene Container konnte nicht gefunden werden.

18

Der Name in der Option CONTAINER enthält ein ungültiges Zeichen oder eine ungültige Kombination von Zeichen.

26

Der in der Option CONTAINER angegebene Prozesscontainer ist schreibgeschützt.

16 INVREQ

RESP2-Werte:

1

Die Option DATATYPE wurde ohne die Option CHANNEL angegeben und es ist kein aktueller Kanal vorhanden (weil an das Programm, das den Befehl abgesetzt hat, kein Kanal übergeben wurde). Die Option DATATYPE ist nur für Befehle PUT CONTAINER gültig, die einen Kanal (explizit oder implizit) angeben. Sie ist für Befehle PUT CONTAINER (BTS) nicht gültig.

2

Die Option FROMCCSID wurde ohne die Option CHANNEL angegeben und es ist kein aktueller Kanal vorhanden (weil an das Programm, das den Befehl abgesetzt hat, kein Kanal übergeben wurde). Die Option FROMCCSID ist nur für Befehle PUT CONTAINER gültig, die einen Kanal (explizit oder implizit) angeben. Sie ist für Befehle PUT CONTAINER (BTS) nicht gültig.

4

Der Befehl wurde außerhalb des Gültigkeitsbereichs einer momentan aktiven Aktivität abgesetzt.

15

Die Option ACQPROCESS wurde verwendet, jedoch hat die Arbeitseinheit, die die Anforderung abgesetzt hat, keinen Prozess angefordert.

24

Die Option ACQACTIVITY wurde verwendet, jedoch hat die Arbeitseinheit, die die Anforderung abgesetzt hat, keine Aktivität angefordert.

25

Die Option PROCESS wurde verwendet, jedoch wurde der Befehl außerhalb des Gültigkeitsbereichs eines momentan aktiven Prozesses abgesetzt.

17 IOERR

RESP2-Werte:

30

Es ist ein Ein-/Ausgabefehler für die Repository-Datei aufgetreten.

31

Der Datensatz in der Repository-Datei wird gerade verwendet.

100 LOCKED

Die Anforderung kann nicht ausgeführt werden, weil eine Sperre für den relevanten Datensatz in der Repository-Datei besteht.

106 PROCESSBUSY

RESP2-Werte:

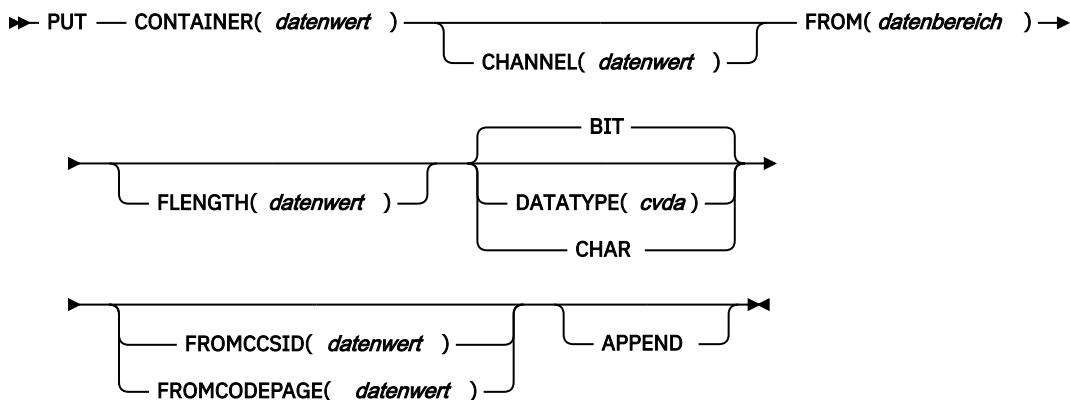
13

Die Anforderung konnte nicht erfüllt werden, weil der Prozessdatensatz durch eine andere Task gesperrt ist.

PUT CONTAINER (CHANNEL)

Speichert Daten in einem angegebenen Kanalcontainer.

PUT CONTAINER (CHANNEL)



Bedingungen: CCSIDERR, CHANNELERR, CODEPAGEERR, CONTAINERERR, INVREQ, LENGERR

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl PUT CONTAINER (CHANNEL) platziert Daten in einem Container, der einem angegebenen Kanal zugeordnet ist.

Der Container wird durch den Namen angegeben.

Der Kanal, der Eigner des Containers ist, kann explizit durch Angabe der Option CHANNEL oder implizit durch Weglassen der Option CHANNEL identifiziert werden. Wenn diese Option nicht angegeben wird, wird der aktuelle Kanal impliziert. Wenn kein aktueller Kanal vorhanden ist, wird der Befehl als Befehl EXEC CICS PUT CONTAINER (BTS) interpretiert. Weitere Informationen finden Sie unter „[PUT CONTAINER \(BTS\)](#)“ auf Seite 353.

Wenn der angegebene Container nicht vorhanden ist, wird er erstellt. Ist der angegebene Container vorhanden, wird sein vorheriger Inhalt überschrieben, sofern Sie nicht die Option APPEND angeben. Wenn der angegebene Kanal nicht vorhanden ist, wird er erstellt.

Es gibt keine Begrenzung für die Anzahl von Containern, die einem Kanal zugeordnet werden können. Die Größe einzelner Container wird nur durch die verfügbare Speicherkapazität begrenzt.



Vorsicht: Wenn Sie mehrere große Container erstellen, schränken Sie möglicherweise die Speicherkapazität ein, die anderen Anwendungen zur Verfügung steht.

Optionen

APPEND

Gibt an, dass die Daten, die an den Container übergeben werden, an die in dem Container vorhandenen Daten angehängt werden. Wenn diese Option nicht angegeben wird, werden die vorhandenen Daten in dem Container durch die Daten überschrieben, die an den Container übergeben werden.

CHANNEL(datenwert)

Gibt den Namen (1 - 16 Zeichen) des Kanals an, der Eigner des Containers ist. Zulässige Zeichen: A-Z a-z 0-9 \$ @ # / % ? ! : | " = , ; > . - und _. Führende und eingebettete Leerzeichen sind nicht zulässig. Wenn der angegebene Name kürzer als 16 Zeichen ist, wird er mit nachfolgenden Leerzeichen bis auf 16 Zeichen aufgefüllt.

Kanalnamen haben immer EBCDIC-Format. Der Satz an zulässigen Zeichen für Kanalnamen, wie zuvor aufgeführt, umfasst einige Zeichen, die nicht in allen EBCDIC-Codepages dieselbe Darstellung haben. Wenn Kanäle zwischen Regionen übertragen werden sollen, wird daher empfohlen, die Zeichen zur Kanalbenennung auf die folgenden zu begrenzen: A-Z a-z 0-9 : = , ; > . - und _.

Wenn der Kanal nicht vorhanden ist, wird er erstellt. Dieser neue Kanal bleibt im Bereich, bis die Verbindungsebene geändert wird. Weitere Informationen zum Kanalbereich finden Sie unter [Geltungsbereich eines Kanals](#).

Sie können den Kanalnamen DFHTRANSACTION angeben, um einen Transaktionskanal zu verwenden. Ein Transaktionskanal verlässt den Bereich nicht, wenn die Verbindungsebene geändert wird: er ist in der Transaktion immer zugänglich. Weitere Informationen finden Sie unter [Kanäle und Container](#).

CONTAINER(datenwert)

Gibt den Namen (1 - 16 Zeichen) des Containers an, in dem die Daten platziert werden sollen.

Zulässige Zeichen: A-Z a-z 0-9 \$ @ # / % ? ! : | " = , ; > . - und _. Führende und eingebettete Leerzeichen sind nicht zulässig. Wenn der angegebene Name kürzer als 16 Zeichen ist, wird er mit nachfolgenden Leerzeichen bis auf 16 Zeichen aufgefüllt.

Verwenden Sie keine Containernamen, die mit DFH beginnen, es sei denn, Sie werden von CICS dazu aufgefordert.

Containernamen haben immer EBCDIC-Format. Der Satz an zulässigen Zeichen für Containernamen, wie zuvor aufgeführt, umfasst einige Zeichen, die nicht in allen EBCDIC-Codepages dieselbe Darstellung haben. Wenn Container zwischen Regionen übertragen werden sollen, wird daher empfohlen, die Zeichen zur Benennung von Containern auf die folgenden zu begrenzen: A-Z 0-9 : = , ; > . - und _.

DATATYPE(cvda)

Gibt den Datentyp der Daten an, die in den Container zu stellen sind. Diese Option gilt nur für neue Container. Wenn der Container vorhanden ist, wurde sein Datentyp festgelegt, als er erstellt wurde. Dieser Datentyp kann nicht geändert werden. Die CVDA-Werte lauten wie folgt:

BIT

Bitdaten. Die Daten im Container können nicht konvertiert werden. Dies ist der Standardwert, sofern nicht FROMCCSID angegeben wird.

CHAR

Zeichendaten. Die Daten, die im Container gespeichert werden sollen, werden (sofern erforderlich) entsprechend der Einstellung im Wert FROMCCSID oder FROMCODEPAGE konvertiert. Wenn die Optionen FROMCCSID und FROMCODEPAGE nicht angegeben werden, wird angenommen, dass die Daten in der CCSID der Region codiert sind, die im Systeminitialisierungsparameter **LOCALCCSID** angegeben ist.

Alle Daten in einem Container werden so konvertiert, als wären sie eine einzige Zeichenfolge. Für SBCS-Codepages ist eine Struktur, die aus mehreren Zeichenfeldern besteht, mit einer Einzelbytezeichenfolge äquivalent. Für DBCS-Codepages ist dies jedoch nicht der Fall. Wenn Sie DBCS-Codepages verwenden, müssen Sie jede Zeichenfolge in einen separaten Container einfügen, um sicherzustellen, dass die Datenkonvertierung ordnungsgemäß funktioniert.

FLENGTH(datenwert)

Gibt die Länge des Datenbereichs, aus dem Daten gelesen werden, als Vollwort-Binärwert an.

FROM(datenbereich)

Gibt einen Datenbereich an, aus dem Daten in den Container geschrieben werden.

FROMCCSID(datenwert)

Gibt die aktuelle ID des codierten Zeichensatzes (Coded Character Set Identifier - CCSID) der Zeichendaten, die in den Container eingefügt werden sollen, als Vollwort-Binärwert an. Wenn Sie es bevorzugen, einen IANA-Namen für die Codepage anzugeben, oder wenn Sie es bevorzugen, die CCSID in Form alphanumerischer Zeichen anzugeben, verwenden Sie stattdessen die Option FROMCODEPAGE. Verwenden Sie diese Option, wenn die Daten, die in den Container eingefügt werden sollen, nicht in der CCSID der Region codiert sind, wie sie im Systeminitialisierungsparameter LOCALCCSID angegeben ist.

Wenn die Option FROMCCSID angegeben wird, wird DATATYPE(DFHVALUE(CHAR)) impliziert.

FROMCODEPAGE(datenwert)

Gibt einen IANA-registrierten alphanumerischen Zeichensatznamen oder eine ID des codierten Zeichensatzes (CCSID) für die aktuelle Codepage der Zeichendaten an, die in den Container eingefügt werden sollen, wobei bis zu 40 alphanumerische Zeichen mit geeigneter Interpunktion verwendet werden. Verwenden Sie diese Option anstelle der Option CCSID, wenn Sie einen von IANA registrierten Zeichensatznamen verwenden wollen, was im Header 'Content-Type' für eine HTTP-Anforderung angegeben wird. CICS konvertiert den IANA-Namen in eine CCSID. Der nachfolgende Datenkonvertie-

rungsprozess ist identisch. Verwenden Sie diese Option außerdem dann, wenn Sie es bevorzugen, die CCSID in alphanumerischen Zeichen und nicht als Vollwort-Binärwert anzugeben.

Wenn die Option FROMCODEPAGE angegeben wird, wird DATATYPE(DFHVALUE(CHAR)) impliziert.

Bedingungen

123 CCSIDERR

RESP2-Werte:

1

Die in der Option FROMCCSID angegebene CCSID liegt außerhalb des Bereichs gültiger CCSID-Werte.

2

Die in der Option FROMCCSID angegebene CCSID und die CCSID des Containers sind eine nicht unterstützte Kombination. Die CCSID des Containers ist der Wert, der im ersten Befehl PUT CONTAINER für diesen Container angegeben oder als Standardwert angenommen wurde. Bei der ersten Verwendung einer ungültigen Kombination gibt CICS die Fehlnachricht DFHAP0802 aus, die das Paar der CCSIDs enthält.

4

Mindestens ein Zeichen konnte nicht konvertiert werden. Jedes nicht konvertierte Zeichen wurde durch ein Leerzeichen in den konvertierten Daten ersetzt.

5

Es ist ein interner Fehler in der Codepagekonvertierung eines Containers aufgetreten. Dieser Fehler kann nur auftreten, wenn das Ziel des PUT-Befehls ein vorhandener und von CICS erstellter Container ist.

122 CHANNELERR

RESP2-Werte:

1

Der in der Option CHANNEL angegebene Name enthält ein ungültiges Zeichen oder eine ungültige Zeichenkombination.

3

Entweder der aktuelle Kanal oder der in der Option CHANNEL angegebene Kanal ist schreibgeschützt.

125 CODEPAGEERR

RESP2-Werte:

1

Die in der Option FROMCODEPAGE angegebene Codepage wird nicht unterstützt.

2

Die in der Option FROMCODEPAGE angegebene Codepage und die CCSID des Containers sind eine nicht unterstützte Kombination. Die CCSID des Containers ist der Wert, der im ersten Befehl PUT CONTAINER für diesen Container in der Option FROMCODEPAGE oder FROMCCSID angegeben oder als Standardwert angenommen wurde. Bei der ersten Verwendung einer ungültigen Kombination gibt CICS die Fehlnachricht DFHAP0802 aus, die das Paar der CCSIDs enthält.

4

Mindestens ein Zeichen konnte nicht konvertiert werden. Jedes nicht konvertierte Zeichen wurde durch ein Leerzeichen in den konvertierten Daten ersetzt. Dieser Fehler kann nur auftreten, wenn das Ziel des PUT-Befehls ein vorhandener Container ist.

5

Es ist ein interner Fehler in der Codepagekonvertierung eines Containers aufgetreten. Dieser Fehler kann nur auftreten, wenn das Ziel des PUT-Befehls ein vorhandener und von CICS erstellter Container ist.

110 CONTAINERERR

RESP2-Werte:

18

Der Name in der Option CONTAINER enthält ein ungültiges Zeichen oder eine ungültige Kombination von Zeichen.

Anmerkung: Diese Bedingung kann auftreten, wenn das Attribut CHANNEL nicht festgelegt ist und kein Standardkanal vorhanden ist. Der Befehl wird möglicherweise wie ein Befehl PUT CONTAINER (BTS) interpretiert und an den BTS-Benennungsregeln geprüft.

16 INVREQ

RESP2-Werte:

1

Die Option DATATYPE wurde ohne die Option CHANNEL angegeben und es ist kein aktueller Kanal vorhanden (weil an das Programm, das den Befehl abgesetzt hat, kein Kanal übergeben wurde). Die Option DATATYPE ist nur für Befehle PUT CONTAINER gültig, die einen Kanal (explizit oder implizit) angeben.

2

Die Option FROMCCSID wurde ohne die Option CHANNEL angegeben und es ist kein aktueller Kanal vorhanden (weil an das Programm, das den Befehl abgesetzt hat, kein Kanal übergeben wurde). Die Option FROMCCSID ist nur für Befehle PUT CONTAINER gültig, die einen Kanal (explizit oder implizit) angeben.

4

Die Option CHANNEL wurde nicht angegeben, es ist kein aktueller Kanal vorhanden (weil an das Programm, das den Befehl abgesetzt hat, kein Kanal übergeben wurde) und der Befehl wurde außerhalb des Gültigkeitsbereichs der momentan aktiven BTS-Aktivität abgesetzt.

30

Sie haben versucht, in einen CICS-definierten schreibgeschützten Container zu schreiben.

32

Es wurde ein anderer CVDA-Wert als CHAR oder BIT für DATATYPE angegeben.

33

Es wurde versucht, den Datentyp eines vorhandenen Containers zu ändern.

34

Der Datentyp BIT ist mit einer CCSID ungültig.

22 LENGERR

RESP2-Werte:

1

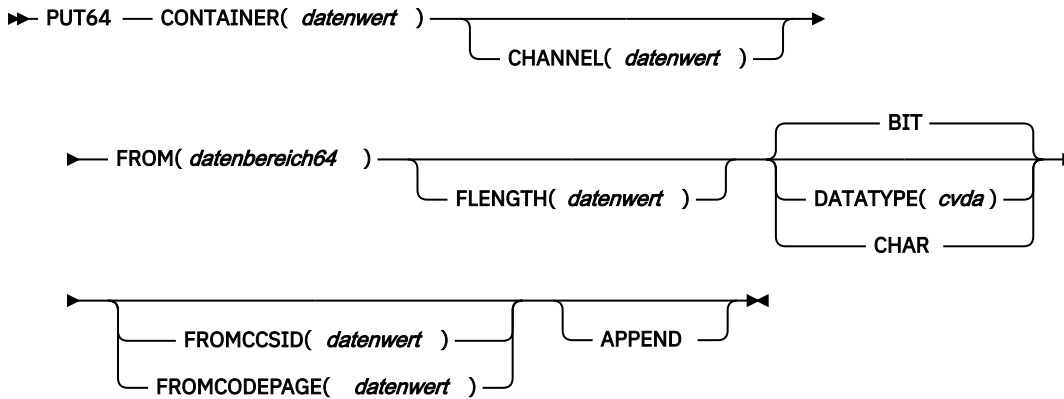
In der Option FLENGTH wurde eine negative Zahl angegeben.

PUT64 CONTAINER

Platziert Daten aus 64-Bit-Speicher in einem benannten Kanalcontainer. Dieser Befehl ist nur zur Verwendung in AMODE(64)-Anwendungsprogrammen in Assemblersprache vorgesehen, die nicht über Language Environment (LE) ausgeführt werden. CICS-Geschäftstransaktionscontainer (BTS-Container) werden nicht unterstützt.

Weitere Informationen finden Sie unter [Beschränkungen und Anforderungen an die Assemblersprachprogrammierung](#).

PUT64 CONTAINER



Bedingungen: CCSIDERR, CHANNELERR, CODEPAGEERR, CONTAINERERR, INVREQ, LENGERR

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl PUT64 CONTAINER stellt Daten aus 64-Bit-Speicher in einen Container, der einem angegebenen Kanal zugeordnet ist.

Der Container wird durch den Namen angegeben. Der Kanal, der Eigner des Containers ist, kann explizit durch Angabe der Option CHANNEL oder implizit durch Weglassen der Option CHANNEL identifiziert werden. Wenn diese Option nicht angegeben wird, wird der aktuelle Kanal impliziert.

Wenn der angegebene Container nicht vorhanden ist, wird er erstellt. Ist der angegebene Container vorhanden, wird sein vorheriger Inhalt überschrieben, sofern Sie nicht die Option APPEND angeben. Wenn der angegebene Kanal nicht vorhanden ist, wird er erstellt.

Es gibt keine Begrenzung für die Anzahl von Containern, die einem Kanal zugeordnet werden können. Die Größe einzelner Container wird nur durch die verfügbare Speicherkapazität begrenzt.

Anmerkung: Beachten Sie, dass Sie möglicherweise die verfügbare Speicherkapazität für andere Anwendungen einschränken, wenn Sie mehrere große Container erstellen.

Optionen

APPEND

Gibt an, dass die Daten, die an den Container übergeben werden, an die in dem Container vorhandenen Daten angehängt werden. Wenn diese Option nicht angegeben wird, werden die vorhandenen Daten in dem Container durch die Daten überschrieben, die an den Container übergeben werden.

CHANNEL(datenwert)

Gibt den Namen (1 - 16 Zeichen) des Kanals an, der Eigner des Containers ist. Zulässige Zeichen: A-Z a-z 0-9 \$ @ # / % ? ! : | " = , ; > . - und _. Führende und eingebettete Leerzeichen sind nicht zulässig. Wenn der angegebene Name kürzer als 16 Zeichen ist, wird er mit nachfolgenden Leerzeichen bis auf 16 Zeichen aufgefüllt.

Kanalnamen haben immer EBCDIC-Format. Der Satz an zulässigen Zeichen für Kanalnamen, wie zuvor aufgeführt, umfasst einige Zeichen, die nicht in allen EBCDIC-Codepages dieselbe Darstellung haben. Wenn Kanäle zwischen Regionen übertragen werden sollen, wird daher empfohlen, die Zeichen zur Kanalbenennung auf die folgenden zu begrenzen: A-Z a-z 0-9 : = , ; > . - und _.

Wenn der Kanal nicht vorhanden ist, wird er erstellt. Dieser neue Kanal bleibt im Bereich, bis die Verbindungsebene geändert wird. Weitere Informationen zum Kanalbereich finden Sie unter [Geltungsbereich eines Kanals](#).

Sie können den Kanalnamen DFHTRANSACTION angeben, um einen Transaktionskanal zu verwenden. Ein Transaktionskanal verlässt den Bereich nicht, wenn die Verbindungsebene geändert wird: er ist in der Transaktion immer zugänglich. Weitere Informationen finden Sie unter [Kanäle und Container](#).

CONTAINER(*datenwert*)

Gibt den Namen (1–16 Zeichen) des Containers an, in dem die Daten platziert werden sollen.

Zulässige Zeichen: A-Z a-z 0-9 \$ @ # / % & ? ! : | " = , ; < > . - und _ . Führende und eingebettete Leerzeichen sind nicht zulässig. Wenn der angegebene Name kürzer als 16 Zeichen ist, wird er mit nachfolgenden Leerzeichen bis auf 16 Zeichen aufgefüllt.

Verwenden Sie keine Containernamen, die mit DFH beginnen, es sei denn, Sie werden von CICS dazu aufgefordert.

Containernamen haben immer EBCDIC-Format. Der Satz an zulässigen Zeichen für Containernamen, wie zuvor aufgeführt, umfasst einige Zeichen, die nicht in allen EBCDIC-Codepages dieselbe Darstellung haben. Wenn Container zwischen Regionen übertragen werden sollen, wird daher empfohlen, die Zeichen zur Benennung von Containern auf die folgenden zu begrenzen: A-Z 0-9 & : = , ; < > . - und _.

DATATYPE(*cvda*)

Gibt den Datentyp der Daten an, die in den Container zu stellen sind. Diese Option gilt nur für neue Container. Wenn der Container vorhanden ist, wurde sein Datentyp festgelegt, als er erstellt wurde. Dieser Datentyp kann nicht geändert werden. Die folgenden CVDA-Werte sind möglich:

BIT

Bitdaten. Die Daten im Container können nicht konvertiert werden. Dies ist der Standardwert, sofern nicht FROMCCSID angegeben wird.

CHAR

Zeichendaten. Die Daten, die im Container gespeichert werden sollen, werden (sofern erforderlich) entsprechend der Einstellung im Wert FROMCCSID oder FROMCODEPAGE konvertiert. Wenn die Optionen FROMCCSID und FROMCODEPAGE nicht angegeben werden, wird angenommen, dass die Daten in der CCSID der Region codiert sind, die im Systeminitialisierungsparameter **LOCALCCSID** angegeben ist.

Alle Daten in einem Container werden so konvertiert, als wären sie eine einzige Zeichenfolge. Für SBCS-Codepages ist eine Struktur, die aus mehreren Zeichenfeldern besteht, mit einer Einzelbytezeichenfolge äquivalent. Für DBCS-Codepages ist dies jedoch nicht der Fall. Wenn Sie DBCS-Codepages verwenden, müssen Sie jede Zeichenfolge in einen separaten Container einfügen, um sicherzustellen, dass die Datenkonvertierung ordnungsgemäß funktioniert.

LENGTH(*datenwert*)

Gibt die Länge des Datenbereichs, aus dem Daten gelesen werden, als Vollwort-Binärwert an.

FROM(*datenbereich64*)

Gibt einen 64-Bit-Datenbereichsverweis auf einen Bereich an, aus dem Daten in den Container geschrieben werden. Der Wert *datenbereich64* bezieht sich auf einen Bereich, auf den durch einen 64-Bit-Zeiger verwiesen wird und der sich im 64-Bit-Speicher (oberhalb der 2-GB-Grenze) befinden kann.

FROMCCSID(*datenwert*)

Gibt die aktuelle ID des codierten Zeichensatzes (Coded Character Set Identifier - CCSID) der Zeichendaten, die in den Container eingefügt werden sollen, als Vollwort-Binärwert an. Wenn Sie es bevorzugen, einen IANA-Namen für die Codepage anzugeben, oder wenn Sie es bevorzugen, die CCSID in Form alphanumerischer Zeichen anzugeben, verwenden Sie stattdessen die Option FROMCODEPAGE. Verwenden Sie diese Option, wenn die Daten, die in den Container eingefügt werden sollen, nicht in der CCSID der Region codiert sind, wie sie im Systeminitialisierungsparameter LOCALCCSID angegeben ist.

Wenn die Option FROMCCSID angegeben wird, wird DATATYPE(DFHVALUE(CHAR)) impliziert.

FROMCODEPAGE(*datenwert*)

Gibt einen IANA-registrierten alphanumerischen Zeichensatznamen oder eine ID des codierten Zeichensatzes (CCSID) für die aktuelle Codepage der Zeichendaten an, die in den Container eingefügt werden sollen, wobei bis zu 40 alphanumerische Zeichen mit geeigneter Interpunktion verwendet werden. Verwenden Sie diese Option anstelle der Option CCSID, wenn Sie einen von IANA registrierten Zeichensatznamen verwenden wollen, was im Header 'Content-Type' für eine HTTP-Anforderung angegeben wird. CICS konvertiert den IANA-Namen in eine CCSID. Der nachfolgende Datenkonvertie-

rungsprozess ist identisch. Verwenden Sie diese Option außerdem dann, wenn Sie es bevorzugen, die CCSID in alphanumerischen Zeichen und nicht als Vollwort-Binärwert anzugeben.

Wenn die Option FROMCCSID angegeben wird, wird DATATYPE(DFHVALUE(CHAR)) impliziert.

Bedingungen

123 CCSIDERR

RESP2-Werte:

- 1**
Die in der Option FROMCCSID angegebene CCSID liegt außerhalb des Bereichs gültiger CCSID-Werte.
- 2**
Die in der Option FROMCCSID angegebene CCSID und die CCSID des Containers sind eine nicht unterstützte Kombination. Die CCSID des Containers ist der Wert, der im ersten Befehl PUT CONTAINER für diesen Container angegeben oder als Standardwert angenommen wurde. Bei der ersten Verwendung einer ungültigen Kombination gibt CICS die Fehlnachricht DFHAP0802 aus, die das Paar der CCSIDs enthält.
- 4**
Mindestens ein Zeichen konnte nicht konvertiert werden. Jedes nicht konvertierte Zeichen wurde durch ein Leerzeichen in den konvertierten Daten ersetzt.
- 5**
Es ist ein interner Fehler in der Codepagekonvertierung eines Containers aufgetreten. Dieser Fehler kann nur auftreten, wenn das Ziel des PUT-Befehls ein vorhandener und von CICS erstellter Container ist.

122 CHANNELERR

RESP2-Werte:

- 1**
Der Name in der Option CHANNEL enthält ein ungültiges Zeichen oder eine ungültige Kombination von Zeichen.
- 3**
Entweder der aktuelle Kanal oder der in der Option CHANNEL angegebene Kanal ist schreibgeschützt.

125 CODEPAGEERR

RESP2-Werte:

- 1**
Die in der Option FROMCODEPAGE angegebene Codepage wird nicht unterstützt.
- 2**
Die in der Option FROMCODEPAGE angegebene Codepage und die CCSID des Containers sind eine nicht unterstützte Kombination. Die CCSID des Containers ist der Wert, der im ersten Befehl PUT CONTAINER für diesen Container in der Option FROMCODEPAGE oder FROMCCSID angegeben oder als Standardwert angenommen wurde. Bei der ersten Verwendung einer ungültigen Kombination gibt CICS die Fehlnachricht DFHAP0802 aus, die das Paar der CCSIDs enthält.
- 4**
Mindestens ein Zeichen konnte nicht konvertiert werden. Jedes nicht konvertierte Zeichen wurde durch ein Leerzeichen in den konvertierten Daten ersetzt. Dieser Fehler kann nur auftreten, wenn das Ziel des PUT-Befehls ein vorhandener Container ist.
- 5**
Es ist ein interner Fehler in der Codepagekonvertierung eines Containers aufgetreten. Dieser Fehler kann nur auftreten, wenn das Ziel des PUT-Befehls ein vorhandener und von CICS erstellter Container ist.

110 CONTAINERERR

RESP2-Werte:

18

Der Name in der Option CONTAINER enthält ein ungültiges Zeichen oder eine ungültige Kombination von Zeichen.

16 INVREQ

RESP2-Werte:

1

Die Option DATATYPE wurde ohne die Option CHANNEL angegeben und es ist kein aktueller Kanal vorhanden (weil an das Programm, das den Befehl abgesetzt hat, kein Kanal übergeben wurde). Die Option DATATYPE ist nur für Befehle PUT CONTAINER gültig, die einen Kanal (explizit oder implizit) angeben.

2

Die Option FROMCCSID wurde ohne die Option CHANNEL angegeben und es ist kein aktueller Kanal vorhanden (weil an das Programm, das den Befehl abgesetzt hat, kein Kanal übergeben wurde). Die Option FROMCCSID ist nur für Befehle PUT CONTAINER gültig, die einen Kanal (explizit oder implizit) angeben.

4

Die Option CHANNEL wurde nicht angegeben und es ist kein aktueller Kanal vorhanden (weil an das Programm, das den Befehl abgesetzt hat, kein Kanal übergeben wurde).

30

Sie haben versucht, in einen CICS-definierten schreibgeschützten Container zu schreiben.

32

Es wurde ein anderer CVDA-Wert als CHAR oder BIT für DATATYPE angegeben.

33

Es wurde versucht, den Datentyp eines vorhandenen Containers zu ändern.

34

Der Datentyp BIT ist mit einer CCSID ungültig.

22 LENGERR

RESP2-Werte:

1

In der Option FLENGTH wurde eine negative Zahl angegeben.

QUERY CHANNEL

Zählt die Anzahl der Container, die sich in einem Kanal befinden.

QUERY CHANNEL

➤ QUERY CHANNEL(*datenwert*) — CONTAINERCNT(*datenbereich*) ➤

Bedingungen: CHANNELERR

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl QUERY CHANNEL zählt die Anzahl der Container, die sich in einem angegebenen Kanal befinden.

Sie müssen die Option CHANNEL und den Kanal explizit angeben. Sie können den Befehl QUERY CHANNEL mit jedem Kanal verwenden, einschließlich des Transaktionskanals (DFHTRANSACTION) und Kanälen, die von CICS erstellt wurden. Zum Ermitteln des aktuellen Kanals eines Programms, d. h. des Kanals, mit dem das Programm aufgerufen wurde, verwenden Sie den Befehl EXEC CICS ASSIGN CHANNEL.

Optionen

CHANNEL(datenwert)

Gibt den 1 – 16 Zeichen langen Namen des Kanals an.

CONTAINERCNT(datenbereich)

Gibt einen Wert für die Anzahl der Container in dem angegebenen Kanal als Vollwort-Binärwert zurück.

Bedingungen

122 CHANNELERR

RESP2-Werte:

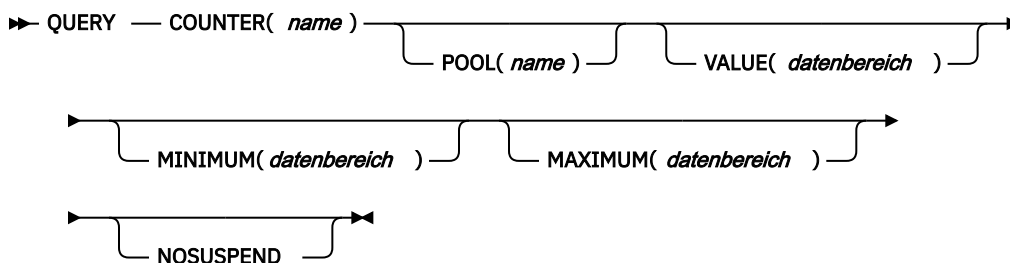
2

Der in der Option CHANNEL angegebene Kanal konnte nicht gefunden werden.

QUERY COUNTER und QUERY DCOUNTER

Frägt einen benannten Zähler ab. COUNTER wird für Vollwortzähler mit Vorzeichen verwendet, DCOUNTER für Doppelwortzähler ohne Vorzeichen.

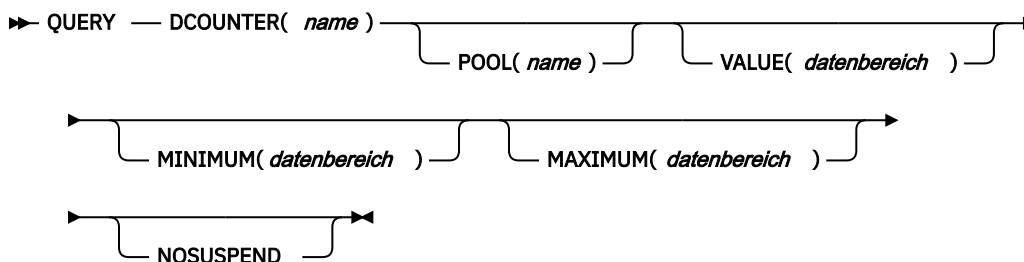
QUERY COUNTER



Bedingungen: BUSY, INVREQ, LENGERR

Dieser Befehl ist threadsicher.

QUERY DCOUNTER



Bedingungen: BUSY, INVREQ

Dieser Befehl ist threadsicher.

Beschreibung

Diese Zählerbefehle geben den aktuellen, den maximalen und den minimalen Wert für den benannten Zähler zurück.

Informationen zur Angabe von Vollwort- und Doppelwortvariablen bei diesen Befehlen für benannte Zähler finden Sie unter [„Argumentwerte für CICS-Befehle“](#) auf Seite 2.

Optionen

COUNTER(name)

Gibt den 16 Zeichen langen Namen des Vollwort-Zählers an, der abgefragt wird. Gültige Zeichen für Namen: A bis Z, 0 bis 9, \$ @ # und _ (Unterstrichzeichen). Wenn *name* eine Variable ist, die einen Namen enthält, der weniger als 16 Zeichen lang ist, muss der Name mit nachfolgenden Leerzeichen aufgefüllt werden.

DCOUNTER(name)

Gibt den 16 Zeichen langen Namen des Doppelwort-Zählers an, der abgefragt wird. Gültige Zeichen für Namen: A bis Z, 0 bis 9, \$ @ # und _ (Unterstreichungszeichen). Wenn *name* eine Variable ist, die einen Namen enthält, der weniger als 16 Zeichen lang ist, muss der Name mit nachfolgenden Leerzeichen aufgefüllt werden.

MAXIMUM(datenbereich)

Gibt den Datenbereich an, in dem CICS den maximalen Wert für den benannten Zähler zurückgeben soll. CICS gibt einen Vollwort-Binärwert mit Vorzeichen für den Befehl COUNTER und einen Doppelwort-Binärwert ohne Vorzeichen für den Befehl DCOUNTER zurück.

MINIMUM(datenbereich)

Gibt den Datenbereich an, in dem CICS den minimalen Wert für den benannten Zähler zurückgeben soll. CICS gibt einen Vollwort-Binärwert mit Vorzeichen für den Befehl COUNTER und einen Doppelwort-Binärwert ohne Vorzeichen für den Befehl DCOUNTER zurück.

NOSUSPEND

Gibt eine sofortige Rückkehr zum Anwendungsprogramm während einer Neuerstellung der Coupling-Facility-Struktur an. Wenn Sie den Parameter NOSUSPEND nicht angeben, wird die Anforderung so lange wiederholt, bis sie erfolgreich ist. Wenn Sie NOSUSPEND angeben, schlägt die Anforderung sofort fehl und CICS gibt die Bedingung BUSY, RESP2=500 zurück.

POOL(poolname)

Gibt den Namen des Pools an, in dem sich der benannte Zähler befindet.

Gültige Zeichen für die Poolselektorzichenfolge sind A bis Z, 0 bis 9, \$ @ # und _ (Unterstreichungszeichen). Falls es sich bei *name* um eine Variable handelt, die einen weniger als 8 Zeichen langen Namen enthält, muss der Name mit nachgestellten Leerzeichen aufgefüllt werden.

Dieser Parameter ist optional. Wenn Sie den Namen des Pools nicht angeben, wird ein Poolauswahlwert aus 8 Leerzeichen angenommen.

Wenn kein entsprechender Eintrag in der Optionstabelle DFHNCOPT vorhanden ist, verwendet CICS den Namen, der im Systeminitialisierungsparameter NCPLDFT angegeben ist, der den Standardpool für benannte Zähler angibt.

Informationen zum Generieren einer Optionstabelle für benannte Zähler mithilfe des Makros DFHNCO finden Sie unter [Server für benannte Zähler](#).

VALUE(datenbereich)

Gibt den Datenbereich an, in dem CICS den aktuellen Wert für den benannten Zähler zurückgeben soll. CICS gibt einen Vollwort-Binärwert mit Vorzeichen für den Befehl COUNTER und einen Doppelwort-Binärwert ohne Vorzeichen für den Befehl DCOUNTER zurück.

Beachten Sie, dass CICS keine Ausnahmebedingung zurückgibt, wenn sich der benannte Zähler in der Bedingung 'Zähler am Grenzwert' befindet. In diesem Fall gibt CICS eine normale Antwort mit einem Wert, der um 1 größer als der Maximalwert ist, der für den Zähler angegeben oder angenommen wurde, unter Addition ohne Vorzeichen zurück. Wenn der Maximalwert die höchste positive Zahl ist, die in einem Vollwort mit Vorzeichen enthalten sein kann, ist der Wert, der von QUERY COUNTER für eine Bedingung 'Zähler am Grenzwert' zurückgegeben wird, die größte negative Zahl.

Bedingungen**128 BUSY**

RESP2-Werte:

500

Die Option NOSUSPEND wurde im Befehl angegeben und die Coupling-Facility-Struktur ist momentan während einer Neuerstellung nicht verfügbar.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

RESP2-Werte:

201

Der benannte Zähler wurde nicht gefunden.

301

Der Server hat einen Fehlercode gemeldet, der für die Schnittstelle des benannten Zählers unverständlich ist. Dies ist in der Regel nur dann möglich, wenn das Schnittstellenlademodul DFHNCIF eine geringere Wartungsstufe oder ein niedrigeres Release-Level als der Server selbst aufweist.

303

Bei einem Makro, das für den Zugriff auf die Coupling-Facility verwendet wird, trat ein unerwarteter Fehler wie beispielsweise eine Strukturstörung oder eine Verbindungsunterbrechung auf. Weitere Informationen enthält die Nachricht DFHNC0441 im Jobprotokoll der Anwendung.

304

Der im Programm angegebene Parameter für die Poolauswahl kann anhand der aktuellen Optionstabelle nicht in einen gültigen Servernamen aufgelöst werden.

305

Die Schnittstelle kann keine Verbindung zum Server für den ausgewählten Pool mit benannten Zählern herstellen. Weitere Informationen enthält eine AXM-Servicenachricht (AXMSCnnnn) im Jobprotokoll der Anwendung.

306

Während der Server eine Anforderung verarbeitete, trat eine abnormale Beendigung auf. Weitere Informationen enthält eine Nachricht im Jobprotokoll der Anwendung und im Jobprotokoll des Servers.

308

Das Optionstabellenmodul DFHNCOPT, das zum Auflösen eines Poolnamens benötigt wird, konnte nicht geladen werden.

309

Während der Verarbeitung der Optionstabelle hat die Schnittstelle für benannte Zähler ein unbekanntes Eintragsformat festgestellt. Entweder wurde die Optionstabelle nicht ordnungsgemäß generiert oder das Schnittstellenlademodul DFHNCIF hat nicht dasselbe Release-Level wie die Optionstabelle.

310

Ein Optionstabelleneintrag, der mit dem angegebenen Poolnamen übereinstimmt, hat ein Benutzerexitprogramm angegeben, aber das Benutzerexitprogramm ist nicht durch eine Linkbearbeitung mit der Optionstabelle verbunden und kann nicht geladen werden.

311

Eine Antwort vom Server des benannten Zählers an das Schnittstellenmodul DFHNCIF der Clientregion gibt an, dass gegenwärtig ein systemverwalteter erneuter Build ausgeführt wird, aber die Bedingung wird von der EXEC CICS-Schnittstelle nicht erkannt. Dies bedeutet, dass die CICS-Region den Stand von CICS TS 2.1 oder früher aufweist.

403

Der Parameter POOL enthält ungültige Zeichen oder eingebettete Leerzeichen.

404

Der Parameter COUNTER enthält ungültige Zeichen oder eingebettete Leerzeichen.

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

Die Bedingung LENGERR tritt nur für COUNTER-Befehle auf und gilt nicht für DCOUNTER-Befehle. Sie tritt auf, wenn ein Zähler, der durch einen DCOUNTER-Befehl oder durch die CALL-Schnittstelle definiert wurde, einen Wert enthält, der zu groß ist, um als Vollwort-Binärwert ohne Vorzeichen korrekt dargestellt zu werden (d. h., der Zähler verwendet mehr als 31 Bit).

In jedem der drei Fälle von Überlauf schließt der Server des benannten Zählers die Operation ab und gibt eine Warnungsantwort an CICS zurück, die CICS an Ihr Anwendungsprogramm als RESP2-Wert zurückgibt. Der Datenbereich enthält die niedrigstwertigen 32 Bit, die vom Server des benannten Zählers zurückgegeben werden, die eine negative Zahl darstellen könnten.

RESP2-Werte:

001

Der aktuelle Wert, den der Server versucht hat, in einem der Datenbereiche zurückzugeben, ist in ein höchstwertiges Bit (Vorzeichenbit) übergelaufen (d. h., der Wert ist negativ).

Anmerkung: Die Bedingung LENGERR mit RESP2=001 kann für einen benannten Zähler, der sich in der Bedingung 'Zähler am Grenzwert' befindet, nicht auftreten. Wenn die Bedingung 'Zähler am Grenzwert' erreicht wurde, wird der Wert (der negativ sein kann) mit einer normalen Antwort zurückgegeben.

002

Ein Wert ist nur 1 Bit zu groß für einen Vollwort-Datenbereich. In diesem Fall ist der Überlaufwert genau 1.

003

Ein Wert ist um einen Wert größer 1 zu groß für einen Vollwort-Datenbereich.

Standardaktion: Die Task wird abnormal beendet.

QUERY SECURITY

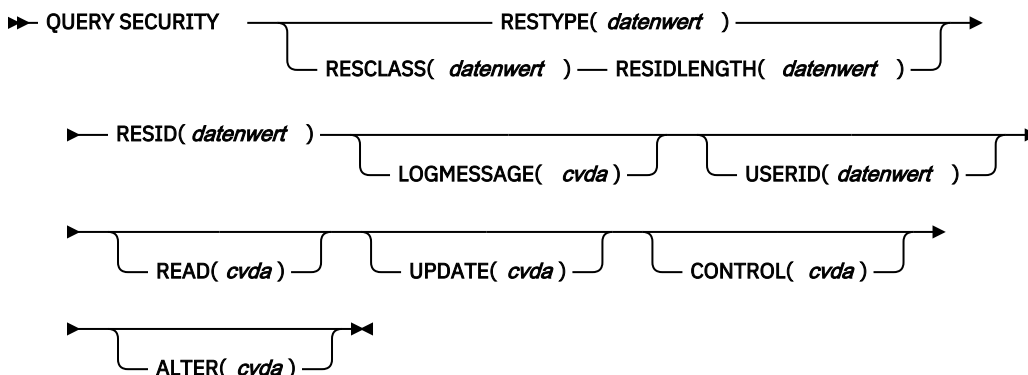
Fragt die Sicherheitsberechtigung eines Benutzers zum Zugriff auf eine Ressource ab. Die Einstellungen RESSEC (Ressourcensicherheit) und CMDSEC (Befehlssicherheit) für eine TRANSACTION-Ressourcendefinition haben auf QUERY SECURITY keine Auswirkungen.

Effekt von SIT-Parametern auf QUERY SECURITY-Befehle

Sie können abfragen, ob ein Benutzer die Berechtigung READ, UPDATE, CONTROL oder ALTER für eine Ressource hat. Welche Werte zurückgegeben werden, hängt davon ab, ob die Sicherheitsprüfung aktiviert ist (**SEC=YES**) und ob die Ressourcensicherheitsprüfung für die relevante Ressourcenklasse aktiviert ist, beispielsweise, ob der Systeminitialisierungsparameter **XFCT=YES** für den Ressourcentyp FILE festgelegt ist.

Tabelle 17. Effekt von SIT-Parametern auf QUERY SECURITY-Befehle					
SIT-Parameter	RACF-Zugriff	cvda aus QUERY SECURITY READ	cvda aus QUERY SECURITY UPDATE	cvda aus QUERY SECURITY CONTROL	cvda aus QUERY SECURITY ALTER
SEC=YES Xnnnn=YES	NONEREADUP- DATECONTROL- ALTER	NOTREADAB- LEREADABLER- EADABLEREA- DABLEREADAB- LE	NOTUPDATAB- LENOTUPDA- TABLEUPDATAB- LEUPDATAB- LEUPDATAB	NOTCTRLABLE- NOTCTRLABLE- NOTCTRLAB- LECTRLAB- LECTRLABLE	NOTALTERABLE- NOTALTERABLE- NOTALTERABLE- NOTALTERABLE- ALTERABLE
SEC=NO Xnnnn=NO	Nicht zutreffend	READABLE	UPDATAB	CTRLABLE	ALTERABLE

QUERY SECURITY



Bedingungen: INVREQ, LENGERR, NOTAUTH, NOTFND, QIDERR, USERIDERR

Dieser Befehl ist threadsicher.

Beschreibung

Mithilfe des Befehls **QUERY SECURITY** kann die Anwendung ermitteln, ob ein Benutzer Zugriff auf die in RACF definierten Ressourcen hat. Diese Ressourcen können sich in CICS-Ressourcenklassen oder in benutzerdefinierten Ressourcenklassen befinden.

Vor dem Aufrufen von RACF überprüft CICS für alle Ressourcen mit Ausnahme von PSBs, ob die Ressource installiert ist. Ist die Ressource nicht vorhanden, ruft CICS RACF nicht auf und gibt die Bedingung NOTFND zurück.

Wenn die Option **USERID** nicht angegeben wird, wird der Benutzer abgefragt, der die Transaktion aufruft, die den Befehl **QUERY SECURITY** absetzt.

Alternativ kann die Anwendung die Sicherheitsberechtigung eines anderen Benutzers abfragen, der in der Option **USERID** angegeben wird.

Weitere Informationen zur Verwendung des Befehls **QUERY SECURITY** finden Sie unter [Sicherheitsprüfung mit Befehl QUERY SECURITY](#).

Optionen

ALTER(cvda)

Fragt ab, ob der Benutzer die Berechtigung **ALTER** für die genannte Ressource hat. Von CICS werden die CVDA-Werte **ALTERABLE** und **NOTALTERABLE** zurückgegeben.

CONTROL(cvda)

Fragt ab, ob der Benutzer die Berechtigung **CONTROL** für die genannte Ressource hat. Von CICS werden die CVDA-Werte **CTRLABLE** und **NOTCTRLABLE** zurückgegeben.

LOGMESSAGE(cvda)

Unterdrückt Nachrichten zu Sicherheitsverstößen. Die an CICS übergebenen Werte sind **LOG** (Standardwert) oder **NOLOG** zur Unterdrückung von Nachrichten.

READ(cvda)

Fragt ab, ob der Benutzer die Berechtigung **READ** für die genannte Ressource hat. Von CICS werden die CVDA-Werte **READABLE** und **NOTREADABLE** zurückgegeben. Die Zugriffsberechtigung **READ** (Lesen) lässt normalerweise eine nicht zerstörerische Verwendung einer Ressource zu, wie zum Beispiel bei den Befehlen **READ** und **INQUIRE**.

RESCLASS(datenwert)

Gibt ein acht Zeichen langes Feld an, das den Namen einer gültigen Ressourcenklasse angibt bei der es sich um eine Nicht-CICS-Klasse in RACF handeln kann. Der Klassenname, der in **RESCLASS** angegeben wird, wird buchstabengetreu ohne Umsetzung verwendet.

Die RACF-Klassen DATASET, GROUP und USER werden in der Klassendeskriptortabelle (Class Descriptor Table - CDT) nicht dargestellt. Dies bedeutet, dass Sie keine Abfragen für diese Klassen durchführen können. Für CICS-Ressourcenklassen muss der Klassenname die Memberklasse und nicht die Gruppenklasse sein; das heißt also CCICSCMD und nicht VCICSCMD.

Wenn es sich bei dem ESM um RACF handelt, kann die Klasse von CICS bereitgestellt werden oder benutzerdefiniert sein. Mit RESCLASS können Sie die abzufragende Berechtigung genauer definieren. Sie können sie zum Beispiel auf Datensatzebene oder Feldebene abfragen.

Um die Ersatzberechtigung eines Benutzers abzufragen, können Sie den Befehl QUERY SECURITY mit der Option RESCLASS('SURROGAT') verwenden. Außerdem müssen Sie die Optionen RESID und RESIDLENGTH angeben. Dieser Befehl wird jedoch **nicht** vom Systeminitialisierungsparameter XUSER gesteuert, daher wird möglicherweise die unerwartete Antwort NOTREADABLE zurückgegeben, wenn Sie XUSER=NO angegeben haben.

Die Antworten, die durch den Befehl zurückgegeben werden, geben die Definition der RESID-Ressource so wieder, wie sie in der in RESCLASS angegebenen Ressourcenklasse definiert ist.

RESID(datenwert)

Gibt den Namen der CICS-Ressource oder der benutzerdefinierten Ressource an, deren Benutzerzugriff Sie abfragen wollen. Der Wert ist eine Zeichenfolge (1 - 12 Zeichen für eine CICS-Ressource und 1 - 246 Zeichen für eine benutzerdefinierte Ressource, sofern Sie nicht die Umsetzeroption COBOL3 verwenden, für die die maximale Länge 160 Zeichen beträgt). Die Angabe von RESID bezieht sich nur dann auf eine von CICS definierte Ressource, wenn RESTYPE('SPCOMMAND') angegeben wird; andernfalls bezieht sie sich auf eine benutzerdefinierte Ressource.

Beim Definieren von RESID-Werten sollten Sie die Auswirkungen der Verwendung von Leerzeichen (X'40 ') in Ressourcenkennungen beachten. Beispiel: In

```
QUERY SECURITY RESTYPE('PSB') RESID('A B')
```

begrenzt das Leerzeichen die RESID und führt dazu, dass RACF den Ressourcennamen A verwendet.

Beachten Sie, dass die tatsächlich geprüfte Ressource davon abhängt, ob RESCLASS oder RESTYPE im Befehl angegeben wird, sowie davon, ob die Voranstellung (Prefixing) aktiv ist (d. h. SECPRFX=YES oder SECPRFX=*präfix* ist als Systeminitialisierungsparameter angegeben).

- Wenn RESCLASS angegeben wird, ist die Ressource, die geprüft wird, immer der tatsächliche RESID-Datenwert, unabhängig davon, ob die Voranstellung aktiv ist oder nicht. Die im Systeminitialisierungsparameter SECPRFX angegebene Voranstellung (Prefixing) gilt nicht für QUERY SECURITY RESCLASS.
- Wenn RESTYPE angegeben wird und die Voranstellung nicht aktiv ist (SECPRFX=NO), ist die Ressource, die geprüft wird, der in RESID angegebene Wert.
- Ansonsten ist die Ressource, die geprüft wird, der Wert in RESID mit der Benutzer-ID der CICS-Region als Präfix (bei SECPRFX=YES) oder mit einem anderen Präfix (bei SECPRFX=*präfix*). Wenn Sie beispielsweise den folgenden Befehl ausgeben, gilt Folgendes:

```
QUERY SECURITY RESTYPE('FILE') RESID('PAYFILE')
```

- Wenn SECPRFX=YES angegeben ist, verwendet CICS die Benutzer-ID der CICS-Region als Präfix und ruft RACF auf, um den Zugriff des Benutzers auf (*benutzer-id_der_cics-region*.PAYFILE zu überprüfen.
- Wenn SECPRFX=*präfix* angegeben ist, verwendet CICS das angegebene Präfix und ruft RACF auf, um den Zugriff des Benutzers auf *präfix*.PAYFILE zu überprüfen.
- Wenn SECPRFX=NO angegeben ist, verwendet CICS kein Präfix und ruft RACF auf, um den Zugriff des Benutzers auf PAYFILE zu überprüfen.

Bei SPCOMMAND werden die Kennungen von CICS vorgegeben. Die folgende Liste gibt die möglichen RESID-Werte für SPCOMMAND an:

Tabelle 18.		
ASSOCIATION ATOMSERVICE AUTINSTMODEL AUTOINSTALL BRFACILITY BUNDLE BUNDLEPART CAPDATAPRED CAPINFOSRCE CAOPTPRED CAPTURESPEC CFDTPOOL CONNECTION CSD DB2CONN DB2ENTRY DB2TRAN DISPATCHER DOCTEMPLATE DSNAME DUMP DUMPDS ENQUEUE EPADAPTER EPADAPTERSET EPADAPTINSET EVENTBINDING EVENTPROCESS EXCI EXITPROGRAM FEPIRESOURCE FILE	HOST IPCONN IRC JOURNALMODEL JOURNALNAME JVMSEVER LIBRARY MODENAME MONITOR MQCONN MQMON MVSTCB NODEJSAPP OSGIBUNDLE OSGISERVICE PARTNER PIPELINE PROCESS PROFILE PROGRAM REQID REQUEST RESETTIME RRMS SECURITY SHUTDOWN STATISTICS STORAGE SUBPOOL SYSDUMPCODE SYSTEM	TASK TCLASS TCPIP TCPIPSERVICE TDQUEUE TEMPSTORAGE TERMINAL TRACEDEST TRACEFLAG TRACETYPE TRANDUMPCODE TRANSACTION TSQUEUE TSMODEL TSPOOL TYPETERM UOW UOWDSNFAIL UOWENQ UOWLINK URIMAP VOLUME VTAM WEB WEBSERVICE XMLTRANSFORM

RESIDLENGTH(*datenwert*)

Gibt die Länge der Ressourcen-ID in RESID als Vollwort-Binärzahl an. Sie können diesen Parameter nur dann verwenden, wenn Sie die Option RESCLASS angeben. Die maximale Länge einer Ressource (RESID) innerhalb einer RACF-Klasse wird in der Klassendescriptortabelle (Class Descriptor Table - CDT) angegeben.

RESTYPE(*datenwert*)

Gibt den Typ von Ressource (1-12 Zeichen) an, auf den Sie den Zugriff des Benutzers abfragen wollen.

Wenn die Ressource nicht bei RACF definiert ist, erteilt CICS keinen Zugriff und die Antwort ist NOT-READABLE. Stellen Sie sicher dass die Länge des mit einer RESTYPE-Anforderung an RACF übergebenen Ressourcennamens die tatsächliche maximale Länge für diesen Ressourcentyp ist.

Der Wert, den Sie für RESTYPE angeben, muss einer der folgenden Ressourcentypen sein:

Tabelle 19. Werte für QUERY SECURITY RESTYPE	
RESTYPE-Wert	Parameter Xname
ATOMSERVICE	XRES
BUNDLE	XRES
DB2ENTRY	XDB2

Tabelle 19. Werte für QUERY SECURITY RESTYPE (Forts.)	
RESTYPE-Wert	Parameter Xname
DOCTEMPLATE	XRES
EPADAPTER	XRES
EPADAPTERSET	XRES
EVENTBINDING	XRES
FILE	XFCT
JOURNALNAME	XJCT
JOURNALNUM (wird aus Gründen der Kompatibilität mit früheren Releases unterstützt)	XJCT
JVMSERVER	XRES
PROGRAM	XPPT
PSB	XPSB
SPCOMMAND (wird zum Angeben einer von CICS definierten Ressource für einen Befehl verwendet)	XCMD
TDQUEUE	XDCT
TRANSACTION	XPCT
TRANSATTACH	XTRAN
TSQUEUE	XTST
TSQNAME	XTST
XMLTRANSFORM	XRES
Nicht zutreffend	XHFS

Der Systeminitialisierungsparameter **XHFS** steuert die Ressourcensicherheit für zFS-Dateien und hat keinen entsprechenden RESTYPE-Wert im Befehl **QUERY SECURITY**. Zugriffssteuerungselemente für zFS-Dateien folgen dem Berechtigungssystem, das von z/OS UNIX System Services verwendet wird, sodass sie anders funktionieren.

Bei dynamischem Transaktionsrouting müssen Sie Transaktionsdefinitionen in Terminalverwaltungsregionen (TOR - Terminal-Owning Regions) nicht installieren. Ein Befehl **QUERY SECURITY** mit dem Wert TRANSATTACH für RESTYPE gibt die Bedingung NOTFND zurück, wenn die Transaktion nicht installiert ist. Anwendungsentwickler müssen sich bewusst sein, dass die Transaktion möglicherweise dynamisch weitergeleitet wird.

Wenn Sie den Befehl **QUERY SECURITY RESTYPE(TRANSATTACH) RESID(tranid) READ(cvda)** absetzen, gibt der Befehl den CVDA-Wert READABLE zurück, wenn der Benutzer über die Berechtigung READ für die Ressource mit dem Namen *tranid* verfügt, und NOTREADABLE, wenn der Benutzer nur über die Berechtigung EXECUTE verfügt. Daher funktionieren Anwendungen nicht, die **QUERY SECURITY RESTYPE(TRANSATTACH)** zum Erstellen eines Menüs der verfügbaren Transaktionen verwenden, wenn die Berechtigung EXECUTE verwendet wird.

Die Antworten, die durch den Befehl zurückgegeben werden, geben die Ergebnisse wieder, die abgerufen würden, wenn ein tatsächlicher Versuch unternommen würde, auf die angegebene CICS-Ressource zuzugreifen.

UPDATE(*cvda*)

Fragt ab, ob der Benutzer die Berechtigung UPDATE für die genannte Ressource hat. Von CICS werden die CVDA-Werte UPDATABLE und NOTUPDATABLE zurückgegeben. Die Zugriffsberechtigung UPDATE (Aktualisieren) lässt normalerweise eine zerstörerische Verwendung einer Ressource zu, wie zum Beispiel bei den Befehlen WRITE, DELETE oder UPDATE.

USERID(*datenwert*)

Gibt die acht Zeichen lange Benutzer-ID des Benutzers an, dessen Zugriff auf die angegebene Ressource abgefragt wird.

Der Benutzer, der die Transaktion aufruft, die den Befehl **QUERY SECURITY** absetzt, muss über die erforderliche Berechtigung zu der Abfrage verfügen, ob ein anderer Benutzer, der in der Option USERID angegeben wird, Zugriff auf die angegebene Ressource hat. CICS führt eine Ersatzbenutzerprüfung durch, um festzustellen, ob der Benutzer, der die Transaktion aufruft, für den Benutzer, der in USERID angegeben wird, berechtigt ist. Schlägt die Ersatzbenutzerprüfung fehl, gibt CICS die Bedingung NOTAUTH zurück.

Bedingungen

16 INVREQ

RESP2-Werte:

7

Der CVDA-Wert ist für die Protokollnachricht (LOGMESSAGE) nicht gültig.

9

Der RESID-Wert ist ungültig oder wurde mit Leerzeichen gefüllt.

10

Der externe Sicherheitsmanager (ESM) ist inaktiv oder nicht vorhanden.

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

RESP2-Werte:

6

Der Wert in RESIDLENGTH ist nicht gültig, das heißt, er liegt nicht im Bereich von 1 bis 246.

Standardaktion: Die Task wird abnormal beendet.

70 NOTAUTH

RESP2-Werte:

102

Die Ersatzbenutzersicherheitsprüfung für die angegebene USERID schlägt fehl.

Die Sicherheitszugriffsfunktionen der Transaktion, die den Befehl abgesetzt hat, lassen die Ausführung des Befehls mit dem in der Option USERID angegebenen Wert nicht zu.

Die Sicherheitszugriffsfunktionen der Transaktion wurden durch den externen Sicherheitsmanager (ESM) entsprechend der Benutzersicherheit sowie abhängig davon eingerichtet, ob die Execution Diagnostic Facility (EDF) verwendet wurde.

Standardaktion: Die Task wird abnormal beendet.

13 NOTFND

RESP2-Werte:

1

Die Ressourcen-ID (RESID) ist nicht gültig.

2

Der Ressourcentyp (RESTYPE) ist nicht gültig.

3

Der RESID-Wert für RESTYPE (SPCOMMAND) ist nicht gültig.

5

Die Ressourcenklasse (RESCLASS) ist für den externen Sicherheitsmanager (ESM) nicht definiert.

8

Die Ressource ist nicht geschützt. Diese Bedingung wird nur bei Verwendung von QUERY SECURITY mit der Option RESCLASS (und nie bei Verwendung mit der Option RESTYPE) zurückgegeben.

Mögliche Ursachen:

- RESCLASS ist nicht aktiv.
- Kein Profil gefunden.
- ESM nicht aktiv.

Standardaktion: Die Task wird abnormal beendet.

44 QIDERR

RESP2-Werte:

1

Ein Name einer indirekten Warteschlange, der der angegebenen Ressourcen-ID (RESID) zugeordnet ist, wurde nicht gefunden.

Standardaktion: Die Task wird abnormal beendet.

69 USERIDERR

RESP2-Werte:

11

Die angegebene Benutzer-ID (USERID) ist dem externen Sicherheitsmanager (ESM) nicht bekannt.

12

Die angegebene Benutzer-ID (USERID) wurde entzogen.

Beispiele für Werte, die von QUERY SECURITY=RESTYPE zurückgegeben werden

SEC=NO

Wenn SEC=NO angegeben ist, gibt der Befehl

```
QUERY SECURITY RESTYPE('FILE') RESID('PAYFILE') ALTER(alter_cvda)
```

Folgendes zurück:

```
alter_cvda = DFHVALUE(ALTERABLE)
```

da SEC=NO bedeutet, dass für die gesamte CICS-Region keine Sicherheitsprüfung durchgeführt wird.

SEC=YES und XFCT=NO

Wenn SEC=YES und XFCT=NO angegeben sind, gibt der Befehl

```
QUERY SECURITY RESTYPE('FILE') RESID('PAYFILE') ALTER(alter_cvda)
```

Folgendes zurück:

```
alter_cvda = DFHVALUE(ALTERABLE)
```

da XFCT=NO bedeutet, dass für Dateien keine Sicherheitsprüfung durchgeführt wird.

SEC=YES, XDCT=YES und SECPRFX=NO

Wenn SEC=YES, XDCT=YES und SECPRFX=NO angegeben sind, gibt der Befehl

```
QUERY SECURITY RESTYPE('TDQUEUE') RESID('TDQ1') READ(read_cvda)
```

Folgendes zurück:

```
read_cvda = DFHVALUE(READABLE)
```

wenn der Benutzer über den Zugriff READ (oder eine höhere Zugriffsstufe) auf 'TDQ1' in der Klasse DCICSDCT oder der Gruppenklasse ECICSDCT verfügt.

SEC=YES, XTRAN=YES und SECPRFX=YES

Wenn SEC=YES, XTRAN=YES und SECPRFX=YES angegeben sind, gibt der Befehl

```
QUERY SECURITY RESTYPE('TRANSATTACH') RESID('TRN1') READ(read_cvda)
```

Folgendes zurück:

```
read_cvda = DFHVALUE(NOTREADABLE)
```

wenn der Benutzer **nicht** über den Zugriff READ (oder eine höhere Zugriffsstufe) auf `benutzer-id_der_cics-region.TRN1` in der Klasse TCICSTRN oder der Gruppenklasse GCICSTRN verfügt.

SEC=YES, XTRAN=YES und SECPRFX=YES

Wenn SEC=YES, XTRAN=YES und SECPRFX=YES angegeben sind, gibt der Befehl

```
QUERY SECURITY RESTYPE('TRANSATTACH') RESID('TRN1') READ(read_cvda)
```

Folgendes zurück:

```
read_cvda = DFHVALUE(NOTREADABLE)
```

wenn der Benutzer **nicht** über den Zugriff READ (oder eine höhere Zugriffsstufe) auf `benutzer-id_der_cics-region.TRN1` in der Klasse TCICSTRN oder der Gruppenklasse GCICSTRN verfügt.

SEC=YES, XCMD=\$USRCMD und SECPRFX=präfix

Wenn SEC=YES, XCMD=\$USRCMD und SECPRFX=*präfix* angegeben sind, gibt der Befehl

```
QUERY SECURITY RESTYPE('TRANSATTACH') RESID('TRN1') READ(read_cvda)
```

Folgendes zurück:

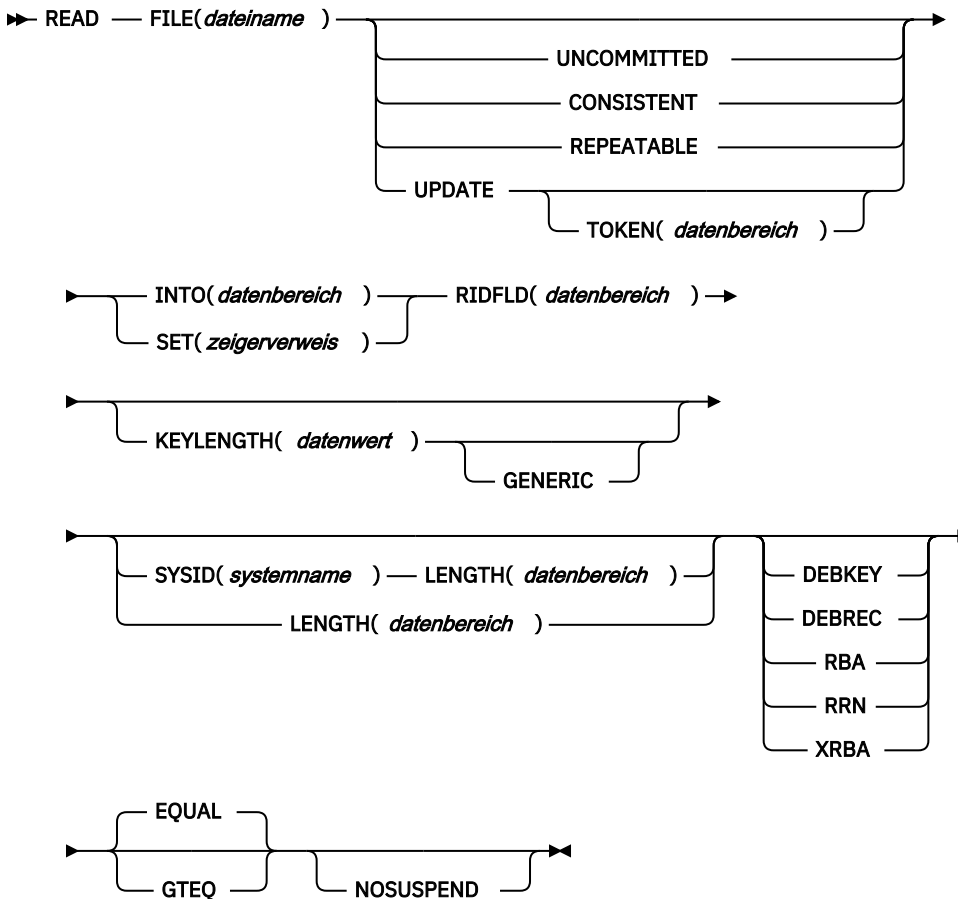
```
read_cvda = DFHVALUE(NOTREADABLE)
```

wenn der Benutzer **nicht** über den Zugriff READ (oder eine höhere Zugriffsstufe) auf `präfix.TRN1` in der Klasse TCICSTRN oder der Gruppenklasse GCICSTRN verfügt.

READ

Liest einen Datensatz aus einer Datei.

READ



Bedingungen: DISABLED, DUPKEY, FILENOTFOUND, ILLOGIC, INVREQ, IOERR, ISCINVREQ, LENGERR, LOADING, LOCKED, NOSPACE, NOTAUTH, NOTFND, NOTOPEN, RECORDBUSY, SYSIDERR

Dieser Befehl ist threadsicher, wenn für die Datei, auf die er sich bezieht, Folgendes zutrifft:

- Sie ist als fern definiert und der Befehl wird über eine IPIC-Verbindung durch Funktionsverlagerung an eine ferne CICS-Region übertragen.
- Sie ist als lokale VSAM-Datei, als VSAM-RLS-Datei oder als Coupling-Facility-Datentabelle definiert.

Dieser Befehl ist nicht threadsicher, wenn für die Datei, auf die er sich bezieht, Folgendes zutrifft:

- Sie ist als fern definiert und der Befehl wird über eine Nicht-IPIC-Verbindung durch Funktionsverlagerung übertragen.
- Sie ist als gemeinsam genutzte Datentabelle oder als BDAM-Datei definiert.

Beschreibung

Der Befehl READ liest einen Datensatz aus einer Datei auf einem lokalen oder fernen System.

Sowohl für den Befehl UPDATE als auch für andere Befehle als UPDATE müssen Sie den abzurufenden Datensatz durch das Datensatz-ID-Feld in der Option RIDFLD angeben. Der RIDFLD-Datenbereich ist sofort nach Abschluss eines Befehls READ UPDATE zur Wiederverwendung durch das Anwendungsprogramm verfügbar.

Hinweise zu Datentabellen

Wenn der Befehl READ eine von CICS verwaltete Datentabelle liest, wird eine READ-Anforderung mit UPDATE oder RBA immer durch einen Aufruf an VSAM erfüllt. Ein Lesevorgang mit vollständigem Schlüssel, der weder ein generischer Lesevorgang noch ein READ UPDATE ist, wird durch einen Verweis auf die Datentabelle erfüllt, sofern dies möglich ist. Wenn der Datensatz in der Tabelle nicht gefunden wird, wird auf die Quellendatei zugegriffen, sofern nicht bekannt ist, dass die Tabelle vollständig ist, das heißt, dass alle Datensätze in der Quelle auch in der Tabelle enthalten sind. (Dies ist der Fall, wenn das Laden abgeschlossen ist und kein Datensatz durch einen Benutzerexit zurückgewiesen wurde.)

Wenn Sie einen generischen Lesevorgang (mit der Option GENERIC) für eine von CICS verwaltete Datentabelle ausführen und CICS eine Bedingung NOTFND zurückgibt, weil der Datensatz in der Tabelle nicht gefunden wurde, löscht CICS den Inhalt der Bereiche INTO() und RIDFLD(), um sicherzustellen, dass kein falscher Datensatz zurückgegeben wird. Dieses Verhalten optimiert die Leistung, unterscheidet sich jedoch von dem Verhalten eines generischen Lesevorgangs für eine VSAM-Datei, bei dem die Bereiche INTO() und RIDFLD() bei einer Bedingung NOTFND unverändert bleiben. Wenn Sie eine VSAM-Datei in eine von CICS verwaltete Datentabelle konvertieren, stellen Sie sicher, dass alle Anwendungen, die generische Lesevorgänge für die Daten durchführen, eine geeignete Aktion ausführen, wenn eine Bedingung NOTFND zurückgegeben wird und der Inhalt der Bereiche INTO() und RIDFLD() gelöscht wird.

Wenn der Befehl READ eine benutzerverwaltete Datentabelle liest, wird nach Abschluss des Ladens nur auf die Datentabelle zugegriffen. Die VSAM-Datei wird in keiner Weise geändert.

Wenn der Befehl READ eine Coupling-Facility-Datentabelle liest, wird nur auf die Datentabelle zugegriffen, auch wenn die Tabelle einleitend aus einer VSAM-Quellendatei geladen wird.

Wenn eine Datei, die sich auf eine benutzerverwaltete Datentabelle oder eine Coupling-Facility-Datentabelle bezieht, mit RLSACCESS(YES) definiert ist, werden die RLS-spezifischen API-Optionen CONSISTENT, NOSUSPEND und REPEATABLE nicht unterstützt.

Dateien durch Zugriff im RLS-Modus lesen

Wenn der Zugriff auf eine Datei im RLS-Modus erfolgt, können Leseanforderungen, die das Schlüsselwort UPDATE nicht angeben, eine der Leseintegritätsoptionen angeben: UNCOMMITTED, CONSISTENT oder REPEATABLE.

Wenn keines dieser Schlüsselwörter angegeben wird, verwendet CICS den Wert, der im Parameter READINTEG der FILE-Ressourcendefinition angegeben ist, für den der Standardwert UNCOMMITTED gilt.

Wenn Sie die Ebene der Leseintegrität verwenden wollen, die im Schlüsselwort READINTEG der FILE-Definition angegeben ist, und von der Verwendung einer lokalen Datei zur Verwendung einer fernen Datei wechseln müssen oder wenn Sie die Position einer fernen Datei ändern, stellen Sie Folgendes sicher:

- Die ferne Dateiverwaltungsregion (FOR - File-Owning Region) unterstützt die Leseintegritätsoptionen.
- Die FILE-Definition im fernen System gibt Folgendes an:
 - RLS-Modus
 - Korrekte Leseintegritätswerte für Ihre Anwendung

READ-Anforderungen, die das Schlüsselwort UPDATE oder die Leseintegritätsoption CONSISTENT oder REPEATABLE (entweder explizit oder implizit in der FILE-Definition) angeben, geben die Bedingung LOCKED zurück, wenn sie auf einen Datensatz verweisen, der eine beibehaltene Sperre hat. Der Schlüssel eines gesperrten Datensatzes wird nicht an das Anwendungsprogramm zurückgegeben. Wenn ein Anwendungsprogramm also GTEQ oder GENERIC in der READ-Anforderung angibt, kann es nicht unterscheiden, welcher Datensatzschlüssel gesperrt ist.

Wenn eine Anforderung, die eine Leseintegrität angibt, durch Funktionsverlagerung an ein Mitglied der CICS-Produktfamilie übertragen wird, das Leseintegrität nicht unterstützt, schlägt die Anforderung fehl:

- Bei Verwendung eines ISC-Links empfängt die Anforderung einen Abbruch mit dem Code ATNI.
- Bei Verwendung eines MRO-Links empfängt die Anforderung einen Abbruch mit dem Code AXF8.

Der Abbruchcode AXF8 gibt an, dass Ihr Programm versucht hat, eine Anforderung, die Dateisteuerungsoptionen angibt, durch Funktionsverlagerung an eine ferne CICS-Region zu übertragen, die diese Optionen nicht unterstützt.

Beibehaltene und aktive Sperren

Die Bedingung RECORDBUSY bezieht sich auf aktive Sperren und die Bedingung LOCKED bezieht sich auf beibehaltene Sperren.

Solche Sperren wirken sich auf READ-Anforderungen aus, die Sperren anfordern, wie zum Beispiel Aktualisierungsanforderungen (UPDATE) und Anforderungen mit Leseintegrität. Dies sind die Arten von READ-Anforderungen, auf die in den folgenden Listenpunkten Bezug genommen wird. Andere READ-Anforderungen bleiben von beibehaltenen oder aktiven Sperren unberührt.

- READ-Anforderungen für Datensätze, die *beibehaltene* Sperren haben, werden immer mit der Antwort LOCKED zurückgewiesen.
- READ-Anforderungen für Datensätze, die *aktive* Sperren haben, warten auf die Freigabe der Sperre, sofern nicht das Schlüsselwort NOSUSPEND angegeben ist. Wenn dies der Fall ist, gibt CICS die Antwort RECORDBUSY zurück.

Optionen

CONSISTENT (nur RLS)

Der Datensatz wird mit der Ebene der Leseintegrität gelesen, die durch eine gemeinsame VSAM-Sperre bereitgestellt wird, die für die Dauer der Anforderung besteht.

Wenn der Datensatz momentan durch eine andere Task geändert wird, die deshalb eine exklusive Sperre besitzt, wartet die READ-Anforderung wie folgt ab, bis die Aktualisierung abgeschlossen ist (sofern nicht auch NOSUSPEND angegeben ist):

- Für eine READ-Anforderung für eine nicht wiederherstellbare Datei wird der Lesevorgang ausgeführt, sobald die VSAM-Anforderung, die die Aktualisierung durchführt, abgeschlossen ist.
- Eine READ-Anforderung für eine wiederherstellbare Datei wird ausgeführt, wenn die Aktualisierungstask ihren nächsten Synchronisationspunkt oder Rollback durchgeführt hat.

DEBKEY

(Geblockte BDAM) Gibt an, dass die Entblockung durch Schlüssel erfolgen soll. Wenn weder DEBREC noch DEBKEY angegeben wird, erfolgt keine Entblockung.

DEBREC

(Geblockte BDAM) Gibt an, dass die Entblockung nach relativem Datensatz (relativ zu null) erfolgen soll. Wenn weder DEBREC noch DEBKEY angegeben wird, erfolgt keine Entblockung.

EQUAL

Gibt an, dass die Suche nur dann von einem Datensatz erfüllt wird, wenn er denselben Schlüssel (vollständig oder generisch) aufweist, der auch in der Option RIDFLD angegeben wird.

FILE(dateiname)

Gibt den Namen der Datei an, auf die zugegriffen werden soll.

Wenn SYSID angegeben wird, wird angenommen, dass sich die Datei (Dataset), auf die sich diese Datei bezieht, auf einem fernen System befindet, und zwar unabhängig davon, ob der Name in CICS definiert ist. Ansonsten wird die Ressourcendefinition verwendet, um zu ermitteln, ob sich die Datei (Dataset) auf einem lokalen oder fernen System befindet.

GENERIC

(VSAM KSDS, Pfade und Datentabellen) Gibt an, dass der Suchschlüssel ein generischer Schlüssel ist, dessen Länge in der Option KEYLENGTH angegeben wird. Die Suche nach einem Datensatz wird erfüllt, wenn ein Datensatz gefunden wird, der dieselben Anfangszeichen (generischer Schlüssel) wie die angegebenen hat.

GTEQ

(VSAM KSDS, Pfade und Datentabellen) Gibt an, dass, wenn die Suche nach einem Datensatz, der denselben Schlüssel (vollständig oder generisch) wie den hat, der in der Option RIDFLD angegeben wird, nicht erfolgreich ist, der erste Datensatz abgerufen wird, der einen größeren Schlüssel hat.

INTO(datenbereich)

Gibt den Datenbereich an, in den der aus der Datei (Dataset) abgerufene Datensatz geschrieben werden soll.

Wenn INTO angegeben wird, muss LENGTH entweder explizit angegeben werden oder muss als Standardwert aus der Option INTO über den Längenattributverweis in Assemblersprache oder über STG und CSTG in PL/I ermittelt werden können. In der Sprache C muss LENGTH explizit angegeben werden.

KEYLENGTH(datenwert)

Gibt die Länge (Halbwort-Binärwert) des Schlüssels an, der in der Option RIDFLD angegeben wurde, außer bei Angabe der Optionen RBA oder RRN, bei denen der Wert von KEYLENGTH nicht gültig ist. Diese Option muss angegeben werden, wenn GENERIC angegeben wird, und kann immer angegeben werden, wenn ein Schlüssel angegeben wird. Wenn jedoch die Länge, die angegeben wird, von der Länge abweicht, die für die Datei (Dataset) definiert ist, und die Operation nicht generisch ist, tritt die Bedingung INVREQ auf.

Die Bedingung INVREQ tritt auch dann auf, wenn GENERIC angegeben wird und der Wert von KEYLENGTH nicht kleiner als der Wert ist, der in der VSAM-Definition angegeben ist.

Wenn KEYLENGTH(0) zu dem Zweck, den ersten Datensatz in der Datei zu lesen, angegeben wird, muss auch die Option GTEQ angegeben werden. Wenn EQUAL entweder explizit oder standardmäßig mit KEYLENGTH(0) angegeben wird, sind die Ergebnisse der READ-Anforderung unvorhersehbar.

Für ferne Dateien kann der Wert von KEYLENGTH in der FILE-Definition angegeben werden. Wenn KEYLENGTH nicht dort definiert wird und im Anwendungsprogramm nicht angegeben wird und der Schlüssel länger als 4 Zeichen ist, ist der Standardwert 4.

LENGTH(datenbereich)

Gibt die Länge des Datenbereichs, in den der Datensatz einzufügen ist, als Halbwort-Binärwert an. Nach Abschluss des Befehls READ enthält der Parameter LENGTH die tatsächliche Länge des Datensatzes.

Diese Option muss angegeben werden, wenn SYSID angegeben wird.

Wenn sich die Datei auf einem fernen System befindet, muss der Parameter LENGTH hier nicht festgelegt werden, jedoch muss er in der Dateiressourcendefinition festgelegt sein.

Wenn sich die Datei auf einem lokalen System befindet, muss der Parameter LENGTH für Datensätze mit variabler Länge mithilfe der Option INTO festgelegt werden, jedoch nicht für Datensätze mit fester Länge. Es ist jedoch aus folgenden Gründen ratsam, die Länge für Datensätze mit fester Länge anzugeben:

- Dadurch erfolgt eine Prüfung, ob der Datensatz, der gelesen wird, für den verfügbaren Datenbereich nicht zu lang ist.
- Wenn Datensätze mit fester Länge in einen Bereich eingelesen werden, der länger als der Datensatz ist, auf den zugegriffen wird, wird für Anwendungen in den Sprachen COBOL, C, PL/I und Assembler die Bedingung LENGERR ausgelöst, wenn die Option LENGTH angegeben wurde. Wenn die Option LENGTH nicht explizit angegeben wird, wird die Bedingung LENGERR zurückgegeben, wenn die Länge aus dem Datenbereich abgeleitet werden kann. Wenn die angegebene Länge die Dateisatzlänge überschreitet, verwendet CICS die längere Länge für das Versetzen. Ist der Zielbereich im Anwendungsprogramm nicht groß genug, wird Speicher über den Zielbereich hinaus überschrieben.

Wenn Sie die Option SET angeben, müssen Sie die Option LENGTH nicht angeben.

Wenn in einen Zieldatenbereich eingelesen wird, der länger als Datensatz ist, der gelesen wird, ist der Inhalt des Zieldatenbereichs vom Ende des abgerufenen Datensatzes bis zum Ende des Zieldatensatzes unvorhersehbar.

Wenn Sie die Option INTO angeben, muss das Argument LENGTH auf einen Datenbereich verweisen, der den größten Datensatz angibt, den das Programm akzeptiert. Falls der abgerufene Datensatz länger als der Wert ist, der in der Option LENGTH angegeben wird, wird der Datensatz auf den angegebenen Wert abgeschnitten und die Bedingung LENGERR ausgelöst. In diesem Fall wird der in LENGTH angegebene Datenbereich auf die Länge des Datensatzes vor dem Abschneiden gesetzt.

Beachten Sie, dass ein Dateisteuerungsbefehl, der für einen Datensatz variabler Länge in einer Datei, die auf dem lokalen CICS-System definiert ist, mit der Bedingung LENGERR fehlschlägt, wenn die Länge nicht angegeben wird. Wenn derselbe Befehl jedoch für eine Datei abgesetzt wird, die auf einem fernen System definiert ist, schlägt der Befehl nicht fehl.

NOSUSPEND (nur RLS)

Die Anforderung wartet nicht, wenn der Datensatz von VSAM mit einer aktiven Sperre gesperrt wurde. Dies gilt auch für Datensätze, die infolge eines Deadlocks gesperrt wurden.

Anmerkung: Anforderungen mit der Angabe NOSUSPEND warten mindestens eine Sekunde, bevor CICS die Antwort RECORDBUSY zurückgibt.

RBA

(Nur VSAM KSDS- oder ESDS- Basisdateien oder von CICS verwaltete Datentabellen, jedoch keine Pfade) Gibt an, dass das Datensatz-ID-Feld, das in der Option RIDFLD angegeben wurde, eine relative Byteadresse (RBA) enthält. Diese Option sollte nur verwendet werden, wenn Datensätze aus einer ESDS-Basis gelesen werden oder wenn das Lesen aus einer KSDS-Basis erfolgt, und relative Byteadressen anstelle von Schlüsseln zur Angabe der Datensätze verwendet werden.

Für die folgenden Elemente kann RBA nicht verwendet werden:

- Benutzerverwaltete Datentabellen
- Coupling-Facility-Datentabellen
- Jede KSDS-Datei, die im RLS-Zugriffsmodus geöffnet ist
- KSDS-Dateien, die erweiterte Adressierung verwenden

Darüber hinaus wird empfohlen, RBA nicht für ESDS-Dateien zu verwenden, die mehr als 4 GB an Daten enthalten. (Verwenden Sie stattdessen XRBA.)

REPEATABLE (nur RLS)

Der Datensatz wird mit der Ebene der Leseintegrität gelesen, die durch eine gemeinsame VSAM-Sperre bereitgestellt wird, die für die Dauer der Arbeitseinheit (UOW) besteht, in der die Leseanforderung abgesetzt wird.

Wenn der Datensatz momentan durch eine andere Task geändert wird, die deshalb eine exklusive Sperre besitzt, wartet die READ-Anforderung wie folgt ab, bis die Aktualisierung abgeschlossen ist (sofern nicht auch NOSUSPEND angegeben ist):

- Eine READ-Anforderung für eine wiederherstellbare Datei wird ausgeführt, wenn die aktualisierende Transaktion ihren nächsten Synchronisationspunkt oder Rollback durchgeführt hat.
- Für eine nicht wiederherstellbare Datei wird der Lesevorgang ausgeführt, sobald die VSAM-Anforderung, die die Aktualisierung durchführt, abgeschlossen ist.

Nach Abschluss der READ-Anforderung bleibt der Datensatz für die Task gesperrt, die die READ-Anforderung abgesetzt hat. Andere Tasks können weiterhin den Datensatz lesen, jedoch erhält keine andere Task die Berechtigung, den Datensatz zu aktualisieren, bevor die Task, die die READ-Anforderung abgesetzt hat, ihren nächsten Synchronisationspunkt oder Rollback durchführt.

RIDFLD(datenbereich)

Gibt das Datensatz-ID-Feld an. Der Inhalt kann ein Schlüssel, eine relative Byteadresse oder eine relative Satznummer (für VSAM-Dateien) sein oder ein Blockverweis, ein physischer Schlüssel und ein Entblockungsargument (für BDAM-Dateien) sein. Für eine relative Byteadresse oder eine relative Satznummer muss das Format dieses Felds ein Vollwort-Binärwert sein. Für eine relative Byteadresse kann der Wert von RIDFLD größer oder gleich null sein. Für eine relative Satznummer muss der Wert von RIDFLD größer oder gleich 1 sein, auch wenn die Option GTEQ angegeben wird.

Weitere Informationen zum Definieren des Datensatz-ID-Felds finden Sie unter [BDAM-Datensätze angeben](#) und [VSAM-Datensätze angeben](#).

Der RIDFLD-Datenbereich ist sofort nach Abschluss eines Befehls zur Wiederverwendung durch das Anwendungsprogramm verfügbar, auch wenn UPDATE angegeben wurde.

Stellen Sie sicher, dass die im Wert von RIDFLD angegebene Variable nicht kleiner als die in diesem Befehl angegebene Schlüssellänge (KEYLENGTH) oder, wenn KEYLENGTH nicht angegeben wird, als die Schlüssellänge der Datei, die Sie lesen, ist. Andernfalls sind die Ergebnisse unvorhersehbar.

RRN

(VSAM RRDS) Gibt an, dass das Datensatz-ID-Feld, das in der Option RIDFLD angegeben wird, eine relative Satznummer (RRN - Relative Record Number) enthält. Diese Option sollte nur mit Dateien verwendet werden, die auf Dateien (Datasets) mit relativen Datensätzen verweisen.

SET(zeigerverweis)

Gibt an, dass CICS einen Puffer bereitstellen sollen, in dem der Datensatz gelesen wird, und gibt den Zeigerverweis an, der die Adresse des abgerufenen Datensatzes enthalten soll.

Wenn die Bedingung DUPKEY in Assemblersprache auftritt, wurde das angegebene Register nicht festgelegt. Das angegebene Register kann über DFHEITP1 geladen werden.

Der Zeigerverweis ist bis zum nächsten Befehl READ für dieselbe Datei oder bis zum Abschluss eines entsprechenden Befehls REWRITE, DELETE oder UNLOCK oder bis zum Synchronisationspunkt im Fall eines Befehls READ UPDATE SET gültig. Wenn Sie die Daten in dem Feld, das durch den Zeiger adressiert wird, behalten wollen, müssen sie in Ihren eigenen Bereich versetzt werden.

Wenn dem Anwendungsprogramm DATALOCATION(ANY) zugeordnet ist, kann die Adresse der Daten ober- oder unterhalb der 16-MB-Grenze liegen.

Wenn dem Anwendungsprogramm DATALOCATION(BELOW) zugeordnet ist, liegt die Adresse der Daten unterhalb der 16-MB-Grenze.

Wenn für die ausführende Transaktion TASKDATAKEY (USER) angegeben ist, sind die zurückgegebenen Daten ein Benutzerschlüssel; andernfalls sind sie ein CICS-Schlüssel.

SYSID(systemname)

Gibt den Namen des Systems an, an das die Anforderung gerichtet ist.

Wenn Sie SYSID angeben und sowohl RBA als auch RRN weglassen, müssen Sie auch LENGTH und KEYLENGTH angeben, da diese Werte in der Ressourcendefinition nicht zu finden sind.

TOKEN(datenbereich)

Gibt eine eindeutige ID für diese READ UPDATE-Anforderung als Vollwort-Binärwert an. Dies ist ein Ausgabewert, der von der Dateisteuerung an die anfordernde Task zu dem Zweck zurückgegeben wird, eine nachfolgende Anforderung REWRITE oder DELETE (oder UNLOCK) dem Datensatz zuzuordnen, der auf diese READ UPDATE-Anforderung hin zurückgegeben wird.

TOKEN kann durch Funktionsverlagerung übertragen werden. Wenn jedoch eine Anforderung mit der Angabe TOKEN durch Funktionsverlagerung an ein Mitglied der CICS-Produktfamilie übertragen wird, das dieses Schlüsselwort nicht erkennt, schlägt die Anforderung fehl.

Anmerkung: Die Angabe von TOKEN impliziert eine Aktualisierung (UPDATE).

UNCOMMITTED

Der Datensatz wird ohne Leseintegrität gelesen.

Es wird der aktuelle Wert des Datensatzes zurückgegeben, wie er in VSAM bekannt ist. Es wird kein Versuch unternommen, diese Leseanforderung mit einer gleichzeitigen Aktualisierungsaktivität für denselben Datensatz zu serialisieren. Der Datensatz wird möglicherweise gerade von einer anderen Task aktualisiert und der Datensatz kann später geändert werden, wenn die Aktualisierung nachfolgend zurückgesetzt wird.

UPDATE

Gibt an, dass der Datensatz zum Aktualisieren oder (für VSAM und Datentabellen) zum Löschen abgerufen werden soll. Wenn diese Option nicht angegeben wird, wird von einer schreibgeschützten Operation ausgegangen.

UPDATE garantiert die Leseintegrität. Der zur Sicherstellung der Datenintegrität verwendete Mechanismus hängt vom Typ der Dateiressource ab:

- Für eine VSAM-Datei, auf die im RLS-Modus zugegriffen wird, wird der zu aktualisierende Datensatz vom SMSVSAM-Server gesperrt.
- Für eine VSAM-Datei, auf die ohne RLS-Modus zugegriffen wird, wird der zu aktualisierende Datensatz von CICS gesperrt und darüber hinaus wird das Steuerintervall, das den Datensatz enthält, in exklusiver Steuerung durch VSAM gehalten.
- Für eine VSAM-Datei, auf die ohne RLS-Modus und mit log(UNDO) zugegriffen wird, hält CICS eine Datensatzsperrung, bis die Task einen Synchronisationspunkt durchführt.
- Für eine BDAM-Datei wird der zu aktualisierende Datensatz in exklusiver Steuerung durch BDAM gehalten.
- Für eine benutzerverwaltete Datentabelle wird der zu aktualisierende Datensatz von CICS gesperrt.
- Für eine von CICS verwaltete Datentabelle wird der zu aktualisierende Datensatz von CICS gesperrt und darüber hinaus wird das Steuerintervall, das den Datensatz enthält, in exklusiver Steuerung durch VSAM gehalten. Die VSAM-Steuerintervallsperrung ist erforderlich, weil Änderungen an der Datentabelle in der Quellendatei (Dataset) abgebildet werden, auf die in einem Nicht-RLS-Modus zugegriffen wird.
- Für eine Coupling-Facility-Datentabelle mit dem Sperrmodell wird der zu aktualisierende Datensatz vom Server der Coupling-Facility-Datentabelle gesperrt.
- Für eine Coupling-Facility-Datentabelle mit dem Konfliktmodell werden Datensätze nicht gesperrt, sodass die Datensätze für eine Aktualisierung durch mehr als eine Task gelesen werden können. Wenn ein Datensatz, der zum Aktualisieren durch eine Task gelesen wurde, dann durch eine andere Task geändert wird, wird die erste Task, wenn sie einen Befehl REWRITE oder DELETE absetzt, durch die Ausnahmebedingung CHANGED benachrichtigt. Wenn ein Datensatz, der zum Aktualisieren durch eine Task gelesen wurde, dann durch eine andere Task gelöscht wird, wird die erste Task, wenn sie einen Befehl REWRITE oder DELETE absetzt, durch die Ausnahmebedingung NOTFND benachrichtigt.

Wenn eine andere Task eine Anforderung READ REPEATABLE für denselben Datensatz abgesetzt hat, wird Ihre Anforderung READ UPDATE veranlasst, zu warten, bis die andere Task einen Synchronisationspunkt (SYNCPPOINT) erreicht (sofern Sie nicht NOSUSPEND angegeben haben).

XRBA

Gibt an, dass das Datensatz-ID-Feld, das in der Option RIDFLD angegeben wird, eine erweiterte relative Byteadresse enthält. Diese Option sollte verwendet werden, wenn Datensätze aus einer ESDS-Datei (Dataset) mit erweiterter Adressierung gelesen werden.

Auf KSDS-Dateien kann durch XRBA nicht zugegriffen werden.

Bedingungen

84 DISABLED

RESP2-Werte:

50

Eine Datei ist inaktiviert, weil sie ursprünglich als inaktiviert definiert wurde und seit dem nicht aktiviert wurde.

Eine Datei ist inaktiviert, weil sie durch einen Befehl SET FILE oder CEMT SET FILE inaktiviert wurde.

Standardaktion: Die Task wird abnormal beendet.

15 DUPKEY

RESP2-Werte (VSAM):

140

Ein Datensatz wird über einen Alternativindex mit dem Attribut NONUNIQUEKEY abgerufen und ein weiterer Alternativindexdatensatz mit demselben Schlüssel folgt.

Wenn in Assemblersprache die Option SET verwendet wird, wurde das angegebene Register nicht festgelegt, kann jedoch über DFHEITP1 geladen werden.

Standardaktion: Die Task wird abnormal beendet.

12 FILENOTFOUND

RESP2-Werte:

1

Der Dateiname, der in der Option FILE angegeben wurde, ist für CICS nicht definiert.

Standardaktion: Die Task wird abnormal beendet.

21 ILLOGIC

RESP2-Werte (VSAM):

110

Es tritt ein VSAM-Fehler auf, der nicht in einer der anderen CICS-Antwortkategorien enthalten ist.

Siehe EIBRCODE im EXEC-Schnittstellenblock. Weitere Informationen finden Sie unter [EIB-Felder](#).

Für benutzerverwaltete Datentabellen tritt diese Bedingung nur für eine READ-Anforderung ohne UPDATE beim Laden auf, wenn CICS versucht hat, den Datensatz aus der Quelldatei (Dataset) abzurufen.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

RESP2-Werte:

20

READ ist gemäß der Ressourcendefinition nicht zulässig.

Ein Befehl READ mit der Option UPDATE wurde für eine Datei abgesetzt, bei der Aktualisierungsoperationen laut Ressourcendefinition nicht zulässig sind.

25

Die Optionen KEYLENGTH und GENERIC werden angegeben und die in der Option KEYLENGTH angegebene Länge ist größer oder gleich der Länge eines vollständigen Schlüssels.

26

Die Option KEYLENGTH wird angegeben (die Option GENERIC jedoch nicht) und die angegebene Länge ist nicht gleich der Länge, die für die Datei (Dataset) definiert ist, auf die sich diese Datei bezieht.

28

Nach einem Befehl READ UPDATE ohne TOKEN wurde ein weiterer Befehl READ UPDATE ohne TOKEN für dieselbe Datei ohne einen dazwischen ausgeführten Befehl REWRITE, DELETE ohne Angabe von RIDFLD, UNLOCK oder SYNCPOINT abgesetzt. Diese Bedingung kann in einigen Fällen ausgelöst werden, obwohl der erste Befehl READ UPDATE nicht erfolgreich war, weil er zum Beispiel das Zeitlimit überschritten hat.

40

Für einen BDAM-Schlüssel trat ein Konvertierungsfehler auf.

42

Die Optionen KEYLENGTH und GENERIC werden angegeben und die in der Option KEYLENGTH angegebene Länge ist kleiner als null.

44

Der Befehl entspricht nicht dem Format von READ für eine benutzerverwaltete Datentabelle oder Coupling-Facility-Datentabelle, wenn zum Beispiel RBA angegeben wird.

51

Eine READ-Anforderung für eine KSDS-Datei, auf die im RLS-Modus zugegriffen wird, gibt das Schlüsselwort RBA an. Der RLS-Modus unterstützt keinen Zugriff über relative Byteadresse auf KSDS-Dateien.

52

In einer READ-Anforderung für eine Datei mit einem anderen Zugriff als im RLS-Modus oder für eine Datentabelle, die mit RLSACCESS(YES) angegeben ist, wird CONSISTENT angegeben. CONSISTENT ist nicht zulässig, wenn sich die Datei auf eine Datentabelle bezieht, auch wenn die Dateidefinition RLSACCESS(YES) angibt.

53

In einer READ-Anforderung für eine Datei mit einem anderen Zugriff als im RLS-Modus oder für eine Datentabelle, die mit RLSACCESS(YES) angegeben ist, wird REPEATABLE angegeben. REPEATABLE ist nicht zulässig, wenn sich die Datei auf eine Datentabelle bezieht, auch wenn die Dateidefinition RLSACCESS(YES) angibt.

55

In einer READ-Anforderung für eine Datei mit einem anderen Zugriff als im RLS-Modus oder für eine Datentabelle, die mit RLSACCESS(YES) angegeben ist, wird NOSUSPEND angegeben. NOSUSPEND ist nicht zulässig, wenn sich die Datei auf eine Datentabelle bezieht, auch wenn die Dateidefinition RLSACCESS(YES) angibt.

56

Ein Versuch, eine wiederherstellbare Coupling-Facility-Datentabelle zu aktualisieren ist fehlgeschlagen, weil die aktuelle Arbeitseinheit bereits 1024 wiederherstellbare Coupling-Facility-Datentabellen aktualisiert hat. Sie können nicht mehr als 1024 wiederherstellbare Coupling-Facility-Datentabellen in einer Arbeitseinheit aktualisieren.

57

Ein Versuch, die Datei zu aktualisieren, ist fehlgeschlagen, weil die zugeordnete VSAM-Datei den Verfügbarkeitsstatus (AVAILABILITY) RREPL hat, der Aktualisierungsoperationen mit Ausnahme solcher von Replikationsprogrammen nicht zulässt.

59

XRBA wurde angegeben, jedoch ist die Datei (Dataset) keine erweiterte ESDS-Datei.

Standardaktion: Die Task wird abnormal beendet.

17 IOERR

RESP2-Werte:

120

Während der READ-Operation ist ein E/A-Fehler aufgetreten. Ein E/A-Fehler ist ein ungewöhnliches Ereignis, das nicht durch eine CICS-Bedingung abgedeckt wird.

Für VSAM-Dateien weist IOERR normalerweise auf einen Hardwarefehler hin.

Für benutzerverwaltete Datentabellen tritt diese Bedingung nur für eine READ-Anforderung ohne UPDATE beim Laden auf, wenn CICS versucht hat, den Datensatz aus der Quelldatei (Dataset) abzurufen.

Für eine Coupling-Facility-Datentabelle weist eine Bedingung IOERR auf eine fehlerhafte Antwort hin, die über einen Coupling-Facility-Zugriff zurückgegeben wurde.

Weitere Informationen sind im EXEC-Schnittstellenblock verfügbar. Einzelheiten finden Sie unter [EIB-Felder](#).

Standardaktion: Die Task wird abnormal beendet.

54 ISCINVREQ

RESP2-Werte:

70

Das ferne System gibt einen Fehler an, der keiner bekannten Bedingung entspricht.

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

RESP2-Werte:

10

In einem Befehl wird weder die Option LENGTH noch die Option SET für eine Datei mit Datensätzen variabler Länge oder für eine BDAM-Datei mit variabler Länge oder für Datensätze mit undefiniertem Format angegeben.

11

Die Länge eines Datensatzes, der mit angegebener Option INTO gelesen wurde, überschreitet den Wert, der in der Option LENGTH angegeben wurde. Der Datensatz wird abgeschnitten und der Datenbereich, der in der Option LENGTH angegeben wurde, wird auf die tatsächliche Länge des Datensatzes gesetzt.

13

Es wurde eine falsche Länge für eine Datei mit Datensätzen fester Länge angegeben.

Standardaktion: Die Task wird abnormal beendet.

94 LOADING

RESP2-Werte:

104

Die Anforderung kann nicht erfüllt werden, weil sie für eine Datentabelle abgesetzt wurde, die momentan noch geladen wird. Die Bedingung kann aus einer der folgenden Ursachen ausgelöst werden:

- Der Befehl READ gibt einen Datensatz an, der noch nicht eine Coupling-Facility-Datentabelle geladen wurde. Datensätze können nur gelesen oder geändert werden, während eine CFDT geladen wird, wenn der angeforderte Schlüssel innerhalb des Bereichs liegt, der bereits geladen wurde.

Die Antwort LOADING kann außerdem für eine Coupling-Facility-Datentabelle zurückgegeben werden, bei deren Laden ein Fehler aufgetreten ist. Weitere Informationen dazu, was geschieht, wenn das Laden einer Coupling-Facility-Datentabelle fehlschlägt, finden Sie in der Beschreibung des globalen Benutzerexits XD TLC in [Exits zur Datentabellenverwaltung \(XDTRD, XDTAD und XD TLC\)](#).

- Der Befehl READ gibt die Option UPDATE für eine benutzerverwaltete Datentabelle an. Eine benutzerverwaltete Datentabelle kann nicht während des Ladens geändert werden.
- Der Befehl READ gibt die Optionen GENERIC oder GTEQ für eine benutzerverwaltete Datentabelle an. Während des Ladens einer benutzerverwalteten Datentabelle (UMT) können Sie Leseanforderungen nur mit genauen Schlüsseln verwenden.

Wenn Ihr Anwendungsprogramm persistent oder zu häufig eine Bedingung LOADING feststellt, prüfen Sie, ob dies nicht durch widersprüchliche Dateidefinitionen verursacht wird, die auf dieselbe Datei (Dataset) verweisen.

Standardaktion: Die Task wird abnormal beendet.

100 LOCKED

RESP2-Werte:

106

Es wird ein Versuch unternommen, einen Datensatz mit Angabe des Schlüsselworts UPDATE oder mit (expliziter oder impliziter) Angabe von CONSISTENT oder REPEATABLE zu lesen, jedoch ist der Datensatz durch eine beibehaltene Sperre gesperrt (siehe [„Beibehaltene und aktive Sperren“](#) auf Seite 377).

Die Bedingung LOCKED kann außerdem für eine Anforderung READ UPDATE für eine wiederherstellbare CFDT auftreten, die das Sperrmodell verwendet, wenn der Datensatz, der gelesen wer-

den soll, durch eine beibehaltene Sperre gesperrt ist. Informationen zur Untersuchung beibehaltener Sperren für Datensätze in einer Coupling-Facility-Datentabelle (CFDT) finden Sie unter [Beibehaltene Sperren der Coupling-Facility-Datentabelle](#).

Standardaktion: Die Task wird mit dem Code AEX8 abnormal beendet.

18 NOSPACE

RESP2-Werte:

100

Auf der Direktzugriffseinheit ist kein Speicherbereich für das Hinzufügen von Datensätzen zu einer Datei verfügbar. Dieser Fehler kann bei einer READ-Anforderung an eine RLS-Datei auftreten, wenn derselbe Fehler für eine vorangegangene WRITE-Anforderung an dieselbe Datei ausgegeben wurde.

70 NOTAUTH

RESP2-Werte:

101

Eine Ressourcensicherheitsprüfung ist für FILE(dateiname) fehlgeschlagen.

Standardaktion: Die Task wird abnormal beendet.

13 NOTFND

RESP2-Werte:

80

Ein Versuch, einen Datensatz auf der Basis des angegebenen Sucharguments abzurufen, ist nicht erfolgreich. Für Datentabellen tritt diese Bedingung auf, wenn ein Versuch, einen Datensatz zu lesen, nicht erfolgreich ist, weil es keinen Eintrag mit dem angegebenen Schlüssel in der Datentabelle gibt. Dies bedeutet nicht, dass kein solcher Datensatz in der Quellendatei vorhanden ist (sofern die Tabelle aus einer erstellt wurde). Es ist möglich, dass ein solcher Datensatz vorhanden ist, jedoch entweder beim einleitenden Laden durch den Benutzerexit XDTRD zurückgewiesen wurde oder nachfolgend aus der Datentabelle gelöscht wurde. Für ferne Dateien tritt diese Bedingung auf, wenn ein Versuch, einen Datensatz zu lesen, ohne Angabe der Schlüssellänge (KEYLENGTH) in der Anwendung oder in der Dateidefinition erfolgt und der tatsächliche Schlüssel länger als vier Zeichen ist.

81

XRBA wurde angegeben und der Wert von RIDFLD war größer als 4 GB, jedoch ist die Datei (Dataset) keine erweiterte ESDS-Datei.

Standardaktion: Die Task wird abnormal beendet.

19 NOTOPEN

RESP2-Werte:

60

Die Bedingung NOTOPEN (RESP2 60) wird aus einem der folgenden Gründe zurückgegeben:

- Die angeforderte Datei ist geschlossen (CLOSED) und nicht aktiviert (UNENABLED). Der Status CLOSED, UNENABLED wird erreicht, wenn eine CLOSE-Anforderung für eine geöffnete aktivierte (OPEN ENABLED) Datei empfangen wurde und die Datei nicht mehr im Gebrauch ist. Sie können CLOSED, UNENABLED auch zum Anfangsstatus machen, indem Sie STATUS(UNENABLED) und OPENTIME(FIRSTREF) in der FILE-Ressourcendefinition angeben. (Für BDAM-Dateien verwenden Sie den Parameter FILSTAT von DFHFCT TYPE=FILE.)
- Die angeforderte Datei ist geöffnet (OPEN) und wird von anderen Transaktionen verwendet, jedoch wurde eine CLOSE-Anforderung für die Datei empfangen.
- Ein Befehl READ wird für eine Datei (Dataset) abgesetzt, die infolge eines Befehls SET DSNAME QUIESCED oder IMMQUIESCED in den Quiescemodus (QUIESCED) versetzt wurde oder gerade wird.
- Die angeforderte Datei ist geschlossen (CLOSED) und aktiviert (ENABLED), sodass CICS versucht hat, die Datei im Rahmen der Ausführung der Anforderung zu öffnen. Das Öffnen der Datei ist

aus einem nicht näher bezeichneten Grund fehlgeschlagen. Sie sollten die Konsole auf Nachrichten prüfen, die erläutern, warum das Öffnen der Datei nicht erfolgreich war.

Diese Bedingung tritt nicht auf, wenn die Anforderung für eine geschlossene, nicht aktivierte Datei (CLOSED, DISABLED) abgesetzt wird. In diesem Fall tritt die Bedingung DISABLED auf.

Standardaktion: Die Task wird abnormal beendet.

101 RECORDBUSY

RESP2-Werte:

107

Das Schlüsselwort NOSUSPEND wurde angegeben und der Datensatz ist durch eine aktive Sperre gesperrt (siehe „Beibehaltene und aktive Sperren“ auf Seite 377).

Standardaktion: Die Task wird mit dem Code AEX9 abnormal beendet.

53 SYSIDERR

RESP2-Werte:

130

Die Option SYSID gibt einen Namen an, der weder das lokale System noch ein fernes System ist, das durch eine CONNECTION- oder IPCONN-Definition definiert ist. SYSIDERR tritt außerdem auf, wenn die Verbindung zu dem fernen System zwar bekannt, jedoch nicht verfügbar ist. Im Fall einer IPCONN-Verbindung tritt die Bedingung SYSIDERR auf, wenn die Verbindung bekannt ist, jedoch das lokale oder das ferne System die Dateisteuerungsbefehle nicht unterstützt, die durch Funktionsverlagerung über die IP-Interkonnektivität übertragen werden.

131

Für eine Coupling-Facility-Datentabelle ist die Herstellung der Verbindung zu dem Coupling-Facility-Datentabellenserver fehlgeschlagen. Dies kann bedeuten, dass der Server selbst ausgefallen ist oder dass der Server verfügbar ist, jedoch CICS keine Verbindung zu ihm hergestellt hat.

132

Der Befehl READ wurde für eine Coupling-Facility-Datentabelle abgesetzt, die nicht mehr vorhanden ist, wahrscheinlich aufgrund eines Fehlers der Coupling-Facility. In diesem Fall fällt der Coupling-Facility-Datentabellenserver ebenfalls aus. Informationen zum erneuten Starten eines Coupling-Facility-Datentabellenservers und zum erneuten Laden einer Tabelle finden Sie unter [Server für Coupling-Facility-Datentabellen einrichten und ausführen](#).

Standardaktion: Die Task wird abnormal beendet.

Beispiele

Das folgende Beispiel zeigt, wie ein Datensatz aus einer Datei mit dem Namen MASTER in einen angegebenen Datenbereich eingelesen wird:

```
EXEC CICS READ
      INTO(RECORD)
      FILE('MASTER')
      RIDFLD(ACCTNO)
```

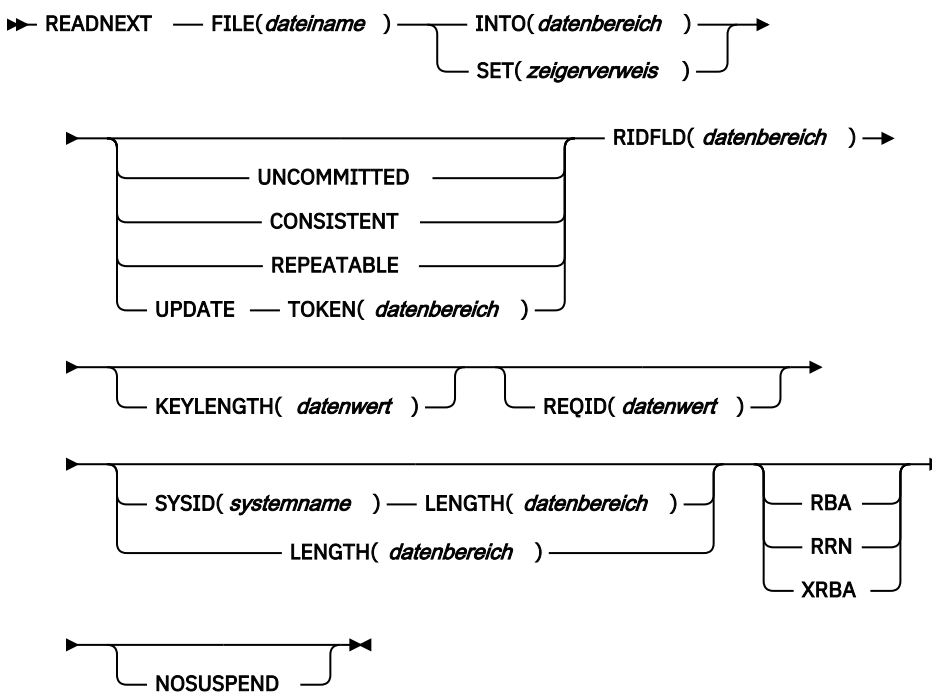
Das folgende Beispiel zeigt, wie ein Datensatz zur Aktualisierung aus einer VSAM-Datei mit einem generischen Schlüssel und unter Angabe einer Suche nach Datensätzen mit größerem oder gleichen Schlüssel gelesen wird.

```
EXEC CICS READ
      INTO(RECORD)
      LENGTH(RECLEN)
      FILE('MASTVSAM')
      RIDFLD(ACCTNO)
      KEYLENGTH(4)
      GENERIC
      GTEQ
      UPDATE
```

Befehl READNEXT

Liest beim Durchsuchen einer Datei den nächsten Datensatz.

Befehl READNEXT



Bedingungen: DUPKEY, ENDFILE, FILENOTFOUND, ILLOGIC, INVREQ, IOERR, ISCINVREQ, LENGERR, LOADING, LOCKED, NOTAUTH, NOTFND, RECORDBUSY, SYSIDERR

Dieser Befehl ist threadsicher, wenn für die Datei, auf die er sich bezieht, Folgendes zutrifft:

- Sie ist als fern definiert und der Befehl wird über eine IPIC-Verbindung durch Funktionsverlagerung an eine ferne CICS-Region übertragen.
- Sie ist als lokale VSAM-Datei, als VSAM-RLS-Datei oder als Coupling-Facility-Datentabelle definiert.

Dieser Befehl ist nicht threadsicher, wenn für die Datei, auf die er sich bezieht, Folgendes zutrifft:

- Sie ist als fern definiert und der Befehl wird über eine Nicht-IPIC-Verbindung durch Funktionsverlagerung übertragen.
- Sie ist als gemeinsam genutzte Datentabelle oder als BDAM-Datei definiert.

Beschreibung

Durch die wiederholte Verwendung des Befehls READNEXT können Datensätze nacheinander aus einer Datei auf einem lokalen oder fernen System gelesen werden. Eine solche Serie von sequenziellen Lesebefehlen wird als *Suchvorgang* in der Datei bezeichnet. Ein Suchvorgang kann auch eine Serie von Befehlen READNEXT und READPREV in beliebiger Reihenfolge umfassen. Ein Suchvorgang muss mit dem Befehl STARTBR initialisiert werden, um den Ausgangspunkt des Suchvorgangs anzugeben; zur Beendigung muss der Befehl ENDBR verwendet werden.

In der Option RIDFLD müssen Sie einen Datenbereich angeben, der groß genug für eine vollständige Kennung von Datensätzen in der Datei ist, also für den vollständigen Schlüssel, die relative Byteadresse (RBA) oder die relative Satznummer (RRN). Dieser Datenbereich kann sowohl als Ausgabeparameter als auch als Eingabeparameter verwendet werden.

Der Datenbereich wird als Ausgabeparameter verwendet, wenn CICS beim Abschluss jedes Befehls READNEXT die vollständige Kennung des soeben abgerufenen Satzes in den Datenbereich der Option RIDFLD stellt. CICS behält anschließend diese Kennung bei, um die Position zu kennzeichnen, an der der nächste Befehl READNEXT fortgesetzt werden muss.

Außer bei BDAM kann der Datenbereich auch als Eingabeparameter verwendet werden. Eine Änderung des Wertes für RIDFLD vor einer Ausgabe des nächsten Befehls READNEXT bewirkt, dass dieser Befehl den Suchvorgang an der neuen Kennung positioniert, von der aus er wie gewohnt fortgesetzt wird. Falls der Suchvorgang mit der Option GENERIC gestartet wurde, muss der geänderte Wert für RIDFLD generisch sein. Wurde der Suchvorgang mit der Option GTEQ gestartet, ist der als nächstes zurückgegebene Datensatz in der Datei der erste Datensatz, dessen Schlüssel größer-gleich dem geänderten Wert von RIDFLD ist.

Ein Befehl READNEXT, der auf einen Befehl READPREV bzw. einen Befehl STARTBR oder RESETBR folgt, bei dem ein 'letzter' Schlüsselwert angegeben war, wird so behandelt, als ob der Wert für RIDFLD geändert worden wäre, und führt (wie oben beschrieben) zu einer Neupositionierung.

Dateien durch Zugriff im RLS-Modus lesen

Für Dateien, auf die im RLS-Modus zugegriffen wird, können Sie das Schlüsselwort UPDATE in die Anforderung READNEXT einbeziehen, um einige Datensätze beim Suchvorgang zu aktualisieren. Bei Angabe von UPDATE müssen Sie ebenfalls TOKEN angeben. Anschließend können Sie den Datensatz durch Ausgabe eines Befehls DELETE oder REWRITE aktualisieren, in dem der von der Suchfunktion zurückgegebene Wert für TOKEN angegeben ist.

Anmerkung: Die Verwendung von TOKEN ohne das Schlüsselwort UPDATE impliziert automatisch die Verwendung von UPDATE.

Für die Verwendung der Option UPDATE gelten die folgenden Regeln:

- Sie können die Option UPDATE bei einem Befehl READNEXT nur dann angeben, wenn auf die Datei im RLS-Modus zugegriffen wird. Falls Sie die Option UPDATE für eine Datei angeben, auf die im Nicht-RLS-Modus zugegriffen wird, gibt CICS den Wert INVREQ zurück.
- Sie können die Option UPDATE bei einem Befehl READNEXT, jedoch nicht bei einem Befehl STARTBR oder RESETBR angeben.
- Sie können Anforderungen mit der Option UPDATE und ohne Aktualisierung in demselben Suchvorgang kombinieren.
- CICS behält die Option UPDATE eines Befehls READNEXT nicht automatisch für den nächsten Befehl bei.

CICS unterstützt in der Sequenz des Suchvorgangs nur einen einzigen Wert für TOKEN; in jedem Befehl READNEXT macht der Wert für TOKEN den vorherigen Wert ungültig.

Sperren für Option UPDATE

Durch die Angabe der Option UPDATE für einen Befehl READNEXT wird eine exklusive Sperre übernommen. Die Dauer dieser exklusiven Sperre innerhalb eines Suchvorgangs ist von der Aktion abhängig, die Ihr Anwendungsprogramm ausführt:

- Falls Sie den letzten Datensatz, der beim Suchvorgang durch einen Befehl READNEXT mit der Option UPDATE unter Verwendung des zugehörigen Tokens angefordert wurde, mit DELETE oder REWRITE löschen bzw. neu schreiben wollen, bleibt die Sperre wie folgt aktiv:
 - Wenn die Datei wiederherstellbar ist, wird die Sperre nach Abschluss des nächsten Synchronisationspunktes oder Rollbacks freigegeben.
 - Wenn die Datei nicht wiederherstellbar ist, wird die Sperre spätestens beim Abschluss von ENDBR, jedoch möglicherweise auch früher freigegeben.
- Falls Sie den letzten gelesenen Datensatz *nicht* aktualisieren, gibt CICS die exklusive Sperre frei, sobald Ihr Programm entweder einen weiteren Befehl READNEXT bzw. READPREV ausgibt oder den Suchvorgang beendet.

Wichtig: Der Befehl UNLOCK bewirkt *keine* Freigabe einer exklusiven Sperre, die von VSAM für einen durch READNEXT UPDATE angeforderten Datensatz gehalten wird. Ein Befehl UNLOCK beim Suchvorgang macht den Wert für TOKEN ungültig, der durch die letzte Anforderung zurückgegeben wurde.

Sperren für Leseintegrität

Die Angabe einer der Optionen für die Leseintegrität übernimmt eine gemeinsame Sperre bei jedem Befehl READNEXT. Die Dauer dieser gemeinsamen Sperren bei einem Suchvorgang ist vom Typ der Leseintegrität abhängig, die Sie angeben:

- Bei Angabe der Leseintegrität CONSISTENT wird die gemeinsame Sperre nur für die Dauer jeder Leseanforderung gehalten, also bis zur Rückgabe des Datensatzes an Ihr Programm.
- Bei Angabe der Leseintegrität REPEATABLE wird die gemeinsame Sperre für die Dauer der Arbeitseinheit gehalten, in der der Suchvorgang ausgeführt wird. In diesem Fall könnte Ihr Programm viele gemeinsame Sperren anfordern, was die Erteilung von exklusiven Sperren für Aktualisierungsfunktionen verhindert. Die Leseintegrität REPEATABLE sollte bei einem Suchvorgang mit Vorsicht verwendet werden.

Funktionsverlagerung bei READNEXT mit UPDATE oder Leseintegrität

Falls ein Befehl READNEXT, der die Option UPDATE oder eine der Leseintegritätsoptionen angibt, per Funktionsverlagerung an ein Mitglied der CICS-Produktfamilie übertragen wird, das die Option UPDATE oder die Leseintegritätsoptionen nicht unterstützt, schlägt die Anforderung fehl:

- Bei Verwendung eines ISC-Links empfängt die Anforderung einen Abbruch mit dem Code ATNI.
- Bei Verwendung einer MRO-Verbindung empfängt die Anforderung einen AXF8-Abbruch.

AXF8 ist ein Abbruchcode, der von der Absenderseite einer Anforderung mit Funktionsverlagerung empfangen wird. Er gibt an, dass versucht wurde, eine Anforderung, in der eine Option UPDATE angegeben ist, über eine MRO-Verbindung an eine CICS-Region zu senden, die Aktualisierungs- oder Leseintegritätsoptionen nicht unterstützt.

Leseintegrität

Wenn im RLS-Modus auf eine Datei zugegriffen wird, können Leseanforderungen ohne Aktualisierungsoption die Leseintegritätsoptionen UNCOMMITTED, CONSISTENT oder REPEATABLE angeben.

Falls Sie keines dieser Schlüsselwörter angeben, verwendet CICS den im Parameter READINTEG der FILE-Ressourcendefinition angegebenen Wert, dessen Standardeinstellung UNCOMMITTED lautet.

Wenn Sie die Leseintegritätsstufe verwenden wollen, die im Schlüsselwort READINTEG der FILE-Definition angegeben ist, und anschließend von der Verwendung einer lokalen Datei zur Verwendung einer fernen Datei wechseln müssen oder die Position einer fernen Datei ändern, muss Folgendes sichergestellt sein:

- Die ferne File-Owning Region (FOR) hat den Stand von CICS Transaction Server for OS/390 Version 1 Release 1 (oder höher).
- Die FILE-Definition auf dem fernen System gibt Folgendes an:
 - RLS-Modus
 - Korrekte Leseintegritätswerte für Ihre Anwendung

Beibehaltene und aktive Sperren

RECORDBUSY bezieht sich auf aktive Sperren, LOCKED bezieht sich auf beibehaltene Sperren.

Diese Sperren wirken sich auf Anforderungen READNEXT aus, die Sperren anfordern, also auf Aktualisierungsanforderungen und Anforderungen mit Leseintegrität. Auf diese Arten von Anforderungen READNEXT wird in den nächsten Listenelementen Bezug genommen. Andere Anforderungen READNEXT werden durch beibehaltene oder aktive Sperren nicht beeinflusst.

- Anforderungen READNEXT für Datensätze mit *beibehaltenen* Sperren werden immer mit einer Antwort LOCKED zurückgewiesen.
- Anforderungen READNEXT für Datensätze mit *aktiven* Sperren warten auf die Freigabe der Sperre. Ausgenommen sind Fälle, bei denen das Schlüsselwort NOSUSPEND angegeben ist; hier gibt CICS die Antwort RECORDBUSY zurück.

Optionen

CONSISTENT (nur RLS)

Der Datensatz wird mit der Leseintegritätsstufe gelesen, die durch eine gemeinsame VSAM-Sperre bereitgestellt wird, die für die Dauer der Anforderung besteht.

Falls der Datensatz durch eine andere Task geändert wird, die daher eine exklusive Sperre hält, wartet die Anforderung READNEXT (außer bei Angabe von NOSUSPEND) folgendermaßen auf den Abschluss der Aktualisierung:

- Bei einer Anforderung READNEXT für eine nicht wiederherstellbare Datei wird der Lesevorgang ausgeführt, sobald eine die Aktualisierung vornehmende VSAM-Anforderung abgeschlossen ist.
- Bei einer Anforderung READNEXT für eine wiederherstellbare Datei wird die Leseanforderung ausgeführt, sobald die Aktualisierungstask ihren nächsten Synchronisationspunkt oder Rollback abschließt.

FILE(*dateiname*)

Gibt den Namen der zu durchsuchenden Datei an.

Bei Angabe eines Wertes für SYSID wird davon ausgegangen, dass sich die Datei, auf die sich dieser Dateiname bezieht, auf einem fernen System befindet, wobei es ohne Belang ist, ob der Name für CICS definiert ist. Ansonsten wird die Ressourcendefinition verwendet, um zu ermitteln, ob sich die Datei (Dataset) auf einem lokalen oder fernen System befindet.

INTO(*datenbereich*)

Gibt den Datenbereich an, in den der aus der Datei abgerufene Datensatz geschrieben werden soll.

KEYLENGTH(*datenwert*)

Gibt die Länge (Halbwort-Binärzahl) des Schlüssels an, der in der Option RIDFLD angegeben ist. Ausgenommen hiervon ist die Angabe der relativen Byteadresse (RBA) oder der relativen Satznummer (RRN); in diesem Fall ist die Option KEYLENGTH nicht gültig.

Falls der Suchvorgang ohne die Option GENERIC gestartet wurde (also ein Suchvorgang mit vollständigem Schlüssel ist) und die angegebene Länge von der für die Datei definierten Länge abweicht, tritt die Bedingung INVREQ auf.

Falls der Suchvorgang mit der Option GENERIC gestartet wurde (also ein Suchvorgang mit generischem Schlüssel ist) und die angegebene Länge größer als die für die Datei angegebene Länge ist, tritt die Bedingung INVREQ auf.

Falls beim letzten Befehl STARTBR oder RESETBR die Optionen GTEQ und GENERIC angegeben waren, wird durch die Ausgabe von READNEXT mit KEYLENGTH(0) der Suchvorgang neu am Beginn der Datei positioniert. Falls die Option EQUAL angegeben war, kann die Wirkung von READNEXT KEYLENGTH(0) nicht vorhergesagt werden.

Bei einem generischen Suchvorgang behält CICS eine aktuelle Schlüssellänge für den Suchvorgang bei. Die aktuelle Schlüssellänge wird mit dem Wert initialisiert, der für KEYLENGTH im Befehl STARTBR angegeben wurde.

Sie können die aktuelle Schlüssellänge ändern, indem Sie bei einem Befehl READNEXT oder RESETBR ein Option KEYLENGTH angeben. Falls die aktuelle Schlüssellänge geändert wird, führt dies zu einer Neupositionierung des Suchvorgangs. Der Suchvorgang wird an demjenigen Schlüssel neu positioniert, dessen Anfangszeichen mit dem Wert übereinstimmen, der in der Option RIDFLD für die aktuelle Schlüssellänge angegeben ist.

Nach einer Anforderung, in der KEYLENGTH(0) angegeben war, ist die aktuelle Schlüssellänge Null.

Falls die Option KEYLENGTH in einem Befehl READNEXT weggelassen wird, bleibt die aktuelle Schlüssellänge gleich und der Suchvorgang wird ohne Neupositionierung fortgesetzt.

Falls bei einem Befehl READNEXT eine Option KEYLENGTH angegeben wird, deren Wert mit der aktuellen Schlüssellänge identisch ist, wird dies nicht als Änderung behandelt und der Suchvorgang nicht neu positioniert. Ausgenommen hiervon ist einzig die Angabe von KEYLENGTH(0), die immer zur einer Neupositionierung des Suchvorgangs am Beginn der Datei führt.

Die Option KEYLENGTH kann bei einem generischen Suchvorgang mit einem Wert angegeben werden, der gleich der vollständigen Schlüssellänge ist. Dies bewirkt keine Änderung der aktuellen Schlüssellänge und keine Neupositionierung des Suchvorgangs. Die Möglichkeit zur Angabe der vollständigen Schlüssellänge bei einem generischen Suchvorgang dient dazu, Anforderungen zuzulassen, die einen Wert für SYSID angeben, um den Umsetzungsprogrammen für die Funktionsverlagerung mitzuteilen, wie lang der Schlüssel ist, damit die Umsetzungsprogramme den Schlüssel in die File-Ownning Region verlagern können.

Ein Suchvorgang kann durch die Änderung des Datenbereichs für RIDFLD neu positioniert werden. Ein generischer Suchvorgang wird nur dann neu positioniert, wenn die Änderung des Datenbereichs für RIDFLD den Teil von RIDFLD ändert, der der aktuellen Schlüssellänge entspricht. Dies hat zur Folge, dass der Suchvorgang nicht durch eine Änderung des Datenbereichs für RIDFLD neu positioniert werden kann, falls die aktuelle Schlüssellänge Null ist.

Für ferne Dateien kann der Wert von KEYLENGTH in der FILE-Definition angegeben werden. Falls die Option KEYLENGTH dort nicht definiert ist, nicht im Anwendungsprogramm angegeben ist und der Schlüssel länger als 4 Zeichen ist, lautet der Standardwert 4.

LENGTH(datenbereich)

Gibt die Länge des Datenbereichs, in den der Datensatz einzufügen ist, als Halbwort-Binärwert an. Beim Abschluss des Befehls READNEXT enthält der Parameter LENGTH die tatsächliche Länge des Datensatzes.

Diese Option muss angegeben werden, wenn SYSID angegeben wird.

Falls sich die Datei auf einem fernen System befindet, muss der Parameter LENGTH hier nicht festgelegt werden, aber in der FILE-Ressourcendefinition festgelegt sein.

Falls sich die Datei auf einem lokalen System befindet, muss der Parameter LENGTH mithilfe der Option INTO für Datensätze mit variabler Länge festgelegt werden, jedoch nicht für Datensätze mit fester Länge. Die Angabe der Länge für Datensätze mit fester Länge ist jedoch aus den folgenden Gründen ratsam:

- Sie bewirkt die Prüfung, ob der gelesene Datensatz nicht zu lang für den verfügbaren Datenbereich ist.
- Bei Suchvorgängen für Datensätze mit fester Länge in einem Bereich, der länger als die Datensätze ist, auf die zugegriffen wird, wird für COBOL-, C- und PL/I-Anwendungen sowie für Anwendungen in Assemblersprache die Bedingung LENGERR ausgelöst, falls die Option LENGTH angegeben ist. Wenn die Option LENGTH nicht explizit angegeben ist, wird die Bedingung LENGERR zurückgegeben, wenn die Länge aus dem Datenbereich abgeleitet werden kann. Überschreitet die angegebene Länge die Länge des Datensatzes, verwendet CICS für das Verschieben die größere Länge. Falls der Zielbereich im Anwendungsprogramm nicht groß genug ist, wird der Speicher über den Zielbereich hinaus überlagert.

Bei Angabe der Option SET müssen Sie nicht ebenfalls die Option LENGTH angeben.

Bei einem Suchvorgang für einen Zieldatenbereich, der länger als der gelesene Datensatz ist, ist der Inhalt des Zieldatenbereichs vom Ende des abgerufenen Datensatzes bis zum Ende des Zieldatenbereichs unvorhersehbar.

Falls Sie die Option INTO angeben, muss das Argument LENGTH ein Datenbereich sein, der den größten vom Programm akzeptierten Datensatz angibt. Falls der abgerufene Datensatz länger als der in der Option LENGTH angegebene Wert ist, wird der Datensatz auf den angegebenen Wert abgeschnitten und die Bedingung LENGERR ausgelöst. In diesem Fall wird der mit LENGTH angegebene Datenbereich auf die Länge des Datensatzes vor dem Abschneiden gesetzt.

Beachten Sie, dass ein Dateisteuerungsbefehl, der für einen Datensatz variabler Länge in einer Datei, die auf dem lokalen CICS-System definiert ist, mit der Bedingung LENGERR fehlschlägt, wenn die Länge nicht angegeben wird. Wird derselbe Befehl jedoch für eine Datei ausgegeben, die auf einem fernem System definiert ist, schlägt der Befehl nicht fehl.

NOSUSPEND (nur RLS)

Die Anforderung wartet nicht, wenn der Datensatz von VSAM mit einer aktiven Sperre gesperrt wurde. Dies gilt auch für Datensätze, die infolge eines Deadlocks gesperrt wurden.

Anmerkung: Anforderungen mit der Angabe NOSUSPEND warten mindestens eine Sekunde, bevor CICS die Antwort RECORDBUSY zurückgibt.

RBA

(Nur VSAM KSDS- oder ESDS- Basisdateien oder von CICS verwaltete Datentabellen, jedoch keine Pfade) Gibt an, dass das Datensatz-ID-Feld, das in der Option RIDFLD angegeben wurde, eine relative Byteadresse (RBA) enthält.

Diese Option muss angegeben sein, wenn die Option RBA im Befehl STARTBR oder RESETBR angegeben war. Sie darf nicht angegeben sein, wenn im Befehl STARTBR oder RESETBR die Option RBA nicht angegeben war.

Die Option RBA kann für Folgendes nicht verwendet werden:

- Benutzerverwaltete Datentabellen
- Coupling-Facility-Datentabellen
- Jede KSDS-Datei, die im RLS-Zugriffsmodus geöffnet ist
- KSDS-Dateien, die erweiterte Adressierung verwenden

Von einer Verwendung der Option RBA wird darüber hinaus bei ESDS-Dateien mit einer Größe von mehr als 4 GB abgeraten. (Verwenden Sie stattdessen XRBA.)

REPEATABLE (nur RLS)

Der Datensatz wird mit der Leseintegritätsstufe gelesen, die durch eine gemeinsame VSAM-Sperre bereitgestellt wird, die für die Dauer der Arbeitseinheit (UOW) besteht, in der die Leseanforderung abgesetzt wird.

Falls der Datensatz durch eine andere Task geändert wird, die daher eine exklusive Sperre hält, wartet die Anforderung READNEXT (außer bei Angabe von NOSUSPEND) folgendermaßen auf den Abschluss der Aktualisierung:

- Bei einer wiederherstellbaren Datei wird die Anforderung READNEXT ausgeführt, sobald die aktualisierende Transaktion ihren nächsten Synchronisationspunkt oder Rollback abschließt.
- Bei einer nicht wiederherstellbaren Datei wird die Anforderung READNEXT ausgeführt, sobald die VSAM-Anforderung, die die Aktualisierung vornimmt, abgeschlossen ist.

Nach Abschluss der Anforderung READNEXT bleibt der Datensatz für die Task gesperrt, die die Anforderung READNEXT ausgegeben hat. Andere Tasks können den Datensatz weiterhin lesen, aber der Datensatz kann von keiner anderen Task aktualisiert werden, bis die Task, die die Anforderung READNEXT ausgegeben hat, ihren nächsten Lesevorgang oder Rollback ausführt.

REQID(datenwert)

Gibt in Form eines Halbwort-Binärwertes eine eindeutige Anforderungskennung für einen Suchvorgang an, mit dessen Hilfe Mehrfachsuchvorgänge für eine Datei gesteuert werden. Falls diese Option nicht angegeben ist, wird vom Standardwert Null ausgegangen.

RIDFLD(datenbereich)

Gibt das Feld für die Satzkennung an. Der Inhalt kann ein Schlüssel, eine relative Byteadresse, eine relative Satznummer (für VSAM-Dateien) oder ein Blockverweis, ein physischer Schlüssel und ein Entblockungsargument (für BDAM-Dateien) sein. Bei einer relativen Byteadresse oder einer relativen Satznummer muss dieses Feld das Format einer Vollwort-Binärzahl besitzen. Für eine relative Byteadresse kann der Wert von RIDFLD größer-gleich 0 sein. Für eine relative Satznummer kann der Wert von RIDFLD größer-gleich 1 sein.

Weitere Informationen zum Definieren des Feldes für die Satzkennung finden Sie unter BDAM-Datensätze angeben und VSAM-Datensätze angeben.

Auch bei einem generischen Suchvorgang sollte diese Option RIDFLD immer groß genug für die vollständige Satzkennung sein, weil das Feld nach Abschluss des Befehls READNEXT von CICS mit der vollständigen Kennung des abgerufenen Datensatzes aktualisiert wird.

RRN

(Schlüsseladressierte VSAM-Dateien) Gibt an, dass das in der Option RIDFLD angegebene Feld für die Satzkennung eine relative Satznummer enthält.

SET(zeigerverweis)

Gibt den Zeigerverweis an, der auf die Adresse des abgerufenen Datensatzes gesetzt werden soll.

Falls bei einer Anwendung in Assemblersprache die Bedingung DUPKEY auftritt, ist das angegebene Register nicht festgelegt worden, kann jedoch aus DFHEITP1 geladen werden.

Der Zeigerverweis ist gültig, bis der nächste Befehl READNEXT oder READPREV die Option SET für denselben Suchvorgang (REQID) und dieselbe Datei angibt. Nach einem Befehl ENDBR oder SYNC-POINT ist der Zeiger nicht mehr gültig. Falls Sie die Daten in dem durch den Zeiger adressierten Feld beibehalten wollen, müssen Sie sie in Ihren eigenen Datenbereich verschieben.

Falls dem Anwendungsprogramm DATALOCATION(ANY) zugeordnet ist, kann die Adresse der Daten ober- oder unterhalb der 16-MB-Grenze liegen.

Falls dem Anwendungsprogramm DATALOCATION(BELOW) zugeordnet ist, liegt die vom SET-Zeiger zurückgegebene Adresse unterhalb der 16-MB-Grenze.

Falls für die aktive Task TASKDATAKEY(USER) angegeben ist, befinden sich die zurückgegebenen Daten im Benutzerschlüsselspeicher, andernfalls befinden sie sich im CICS-Schlüsselspeicher.

SYSID(systemname)

Gibt den Namen des Systems an, an das die Anforderung gerichtet ist.

Falls Sie die Option SYSID angeben und beide Optionen RBA und RRN weglassen, müssen Sie ebenfalls die Optionen LENGTH und KEYLENGTH angeben, weil diese nicht in der Ressourcendefinition zu finden sind.

TOKEN(datenbereich) (nur RLS)

Gibt als Vollwort-Binärwert die Anforderungskennung für diese Anforderung READNEXT UPDATE zurück. Dieser Ausgabewert wird durch die Dateisteuerung an die anfordernde Task zurückgegeben, damit er zur Zuordnung einer nachfolgenden Anforderung REWRITE oder DELETE (bzw. UNLOCK) mit dem für diesen Befehl READNEXT zurückgegebenen Datensatz verwendet werden kann.

Sie müssen den für TOKEN zurückgegebenen Wert bei einem anschließenden Befehl REWRITE oder DELETE angeben, um den Datensatz zu kennzeichnen, der neu geschrieben oder gelöscht wird. Den von CICS für die Option TOKEN zurückgegebenen Wert können Sie auch bei einem nachfolgenden Befehl UNLOCK angeben, um das Token zu kennzeichnen, das ungültig gemacht wird.

Sie müssen die Option TOKEN angeben, wenn Sie die Option UPDATE angeben (bei Angabe der Option TOKEN ohne die Option UPDATE wird jedoch UPDATE angenommen).

CICS unterstützt für einen bestimmten REQID-Wert jeweils nur einen einzigen aktiven Wert für TOKEN. Ein Wert für TOKEN bleibt daher nur bis zum nächsten Befehl READNEXT, READPREV, RESETBR oder ENDBR für diesen Suchvorgang bzw. bis zu einem Befehl REWRITE, DELETE oder UNLOCK gültig (je nachdem, was zuerst eintritt).

TOKEN kann durch Funktionsverlagerung übertragen werden. Falls jedoch eine Anforderung mit Angabe von TOKEN per Funktionsverlagerung an ein Mitglied der CICS-Produktfamilie übertragen wird, das dieses Schlüsselwort nicht erkennt, schlägt die Anforderung fehl.

UNCOMMITTED

Der Datensatz wird ohne Leseintegrität gelesen. Es werden die aktuellen Daten so zurückgegeben, wie sie für VSAM bekannt sind. Es wird nicht versucht, diese Leseanforderung mit einer etwaigen gleichzeitigen Aktualisierungsaktivität für denselben Datensatz zu serialisieren. Es besteht die Möglichkeit, dass der Datensatz gerade durch eine andere Transaktion aktualisiert wird, weshalb sich der

Wert des Datensatzes später ändern kann, falls diese Aktualisierung anschließend zurückgesetzt wird.

UPDATE (nur RLS)

Gibt an, dass der Datensatz zur Aktualisierung oder Löschung abgerufen werden soll. Falls Sie sowohl diese Option als auch die Option TOKEN nicht angeben, wird ein Lesezugriff angenommen.

Bei Angabe der Option UPDATE müssen Sie auch die Option TOKEN angeben.

Die Option UPDATE ist nur für Dateien gültig, die in der lokalen Region definiert sind.

XRBA

Gibt an, dass das in der Option RIDFLD angegebene Feld für die Satzkennung eine erweiterte relative Byteadresse enthält. Diese Option sollte bei Suchvorgängen für Datensätze in einer ESDS-Datei mit erweiterter Adressierung verwendet werden.

Sie können die Option XRBA bei einem Befehl READNEXT nur dann angeben, wenn im zugehörigen Befehl STARTBR oder RESETBR ebenfalls XRBA angegeben war.

Auf KSDS-Dateien kann durch XRBA nicht zugegriffen werden.

Bedingungen

15 DUPKEY

RESP2-Werte (VSAM):

140

Ein Datensatz wird mithilfe eines Alternativindex mit dem Attribut NONUNIQUEKEY abgerufen und es folgt ein weiterer Alternativindexdatensatz mit demselben Schlüssel. Diese Bedingung tritt nicht infolge eines Befehls READNEXT auf, der den letzten der Datensätze mit dem nicht eindeutigen Schlüssel liest.

Falls bei einer Anwendung in Assemblersprache die Option SET verwendet wird, ist das angegebene Register nicht festgelegt worden, kann jedoch aus DFHEITP1 geladen werden.

Standardaktion: Die Task wird abnormal beendet.

20 ENDFILE

RESP2-Werte:

90

Während des Suchvorgangs wurde eine Dateiendebedingung festgestellt.

Standardaktion: Die Task wird abnormal beendet.

12 FILENOTFOUND

RESP2-Werte:

1

Ein in der Option FILE referenzierter Dateiname ist nicht für CICS definiert.

Standardaktion: Die Task wird abnormal beendet.

21 ILLOGIC

Der gegenwärtig ausgeführte Suchvorgang wird beendet, wenn diese Bedingung auftritt.

RESP2-Werte (VSAM):

110

Es ist ein VSAM-Fehler aufgetreten, der zu keiner der anderen CICS-Antwortkategorien gehört.

Weitere Informationen sind bei EIBRCODE im EXEC-Schnittstellenblock verfügbar; Details finden Sie unter [EIB-Felder](#).

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

RESP2-Werte:

20

Die FILE-Definition lässt keine Aktualisierungen zu.

25

Die Option KEYLENGTH wurde für einen generischen Suchvorgang angegeben (also einen Suchvorgang, bei dem GENERIC für den Befehl STARTBR oder den letzten Befehl RESETBR angegeben war) und der Wert von KEYLENGTH war größer als die vollständige Schlüssellänge.

26

Die Option KEYLENGTH wurde für einen nicht generischen Suchvorgang angegeben und die angegebene Schlüssellänge ist nicht mit der Länge identisch, die für die Datei definiert ist, auf die sich diese Dateiangabe bezieht.

34

Der Wert für REQID bzw. SYSID oder der Dateiname stimmt nicht mit dem eines erfolgreichen Befehls STARTBR überein oder es wurde ein Befehl SYNCPOINT ausgegeben, der bewirkt hat, dass die Position von Befehlen STARTBR, READNEXT oder READPREV nicht mehr vorhanden ist.

37

Der Typ der Satzkennung (z. B. Schlüssel oder relative Byteadresse), der für den Zugriff auf eine Datei während des Suchvorgangs verwendet wird, wurde geändert. Die Angabe eines Adressierungstyps bei einem Befehl STARTBR und die eines anderen Typs bei einem Befehl READNEXT ist nicht möglich.

42

Die Option KEYLENGTH wurde für einen generischen Suchvorgang angegeben (also einen Suchvorgang, bei dem GENERIC für den Befehl STARTBR oder den letzten Befehl RESETBR angegeben war) und der Wert von KEYLENGTH war kleiner als Null.

52

Die Leseintegrität CONSISTENT wurde bei einer Anforderung READ für eine Datei ohne RLS-Modus oder für eine Datentabelle angegeben, für die RLSACCESS(YES) gilt. Die Leseintegrität CONSISTENT ist nicht zulässig, falls die Dateiangabe eine Datentabelle referenziert, selbst wenn die FILE-Definition RLSACCESS(YES) angibt.

53

Die Leseintegrität REPEATABLE wurde bei einer Anforderung READ für eine Datei ohne RLS-Modus oder für eine Datentabelle angegeben, für die RLSACCESS(YES) gilt. Die Leseintegrität REPEATABLE ist nicht zulässig, falls die Dateiangabe eine Datentabelle referenziert, selbst wenn die FILE-Definition RLSACCESS(YES) angibt.

54

Die Option UPDATE ist nicht zulässig, weil es sich nicht um eine VSAM-Datei handelt, auf die im RLS-Modus zugegriffen wird.

55

Die Leseintegrität NOSUSPEND wurde bei einer Anforderung READ für eine Datei ohne RLS-Modus oder für eine Datentabelle angegeben, für die RLSACCESS(YES) gilt. Die Leseintegrität NOSUSPEND ist nicht zulässig, falls die Dateiangabe eine Datentabelle referenziert, selbst wenn die FILE-Definition RLSACCESS(YES) angibt.

57

Ein Versuch, die Datei zu aktualisieren, ist fehlgeschlagen, weil die zugeordnete VSAM-Datei den Verfügbarkeitsstatus (AVAILABILITY) RREPL hat, der Aktualisierungsoperationen mit Ausnahme solcher von Replikationsprogrammen nicht zulässt.

Standardaktion: Die Task wird abnormal beendet.

17 IOERR

RESP2-Werte:

120

Während des Befehls READNEXT trat ein E/A-Fehler auf. Ein E/A-Fehler ist ein nicht erwartetes Ereignis, das nicht durch eine CICS-Bedingung abgedeckt ist.

Bei VSAM-Dateien gibt die Bedingung IOERR für gewöhnlich einen Hardwarefehler an.

Bei einer Coupling-Facility-Datentabelle gibt eine Bedingung IOERR an, dass eine falsche Antwort von einem Coupling-Facility-Zugriff zurückgegeben wurde.

Weitere Informationen sind im EXEC-Schnittstellenblock verfügbar; Details finden Sie unter [EIB-Felder](#).

Standardaktion: Die Task wird abnormal beendet.

54 ISCINVREQ

RESP2-Werte:

70

Das ferne System gibt einen Fehler an, der keiner bekannten Bedingung entspricht.

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

RESP2-Werte:

10

Für eine Datei mit Datensätzen variabler Länge oder eine BDAM-Datei mit Datensätzen nicht definierten Formats wurde weder die Option LENGTH noch die Option SET angegeben.

11

Die Länge des mit angegebener Option INTO gelesenen Datensatzes überschreitet den Wert, der in der Option LENGTH angegeben ist; der Datensatz wird abgeschnitten und der in der Option LENGTH bereitgestellte Datenbereich wird auf die tatsächliche Länge des Datensatzes gesetzt.

13

Für eine Datei mit Datensätzen fester Länge wurde eine falsche Länge angegeben.

Standardaktion: Die Task wird abnormal beendet.

94 LOADING

RESP2-Werte:

104

Die Leseanforderung hat einen Satzschlüssel für einen Datensatz in einer Coupling-Facility-Datentabelle angegeben, die noch geladen wird, und der Schlüssel gehört nicht zu dem Bereich der bereits geladenen Datensätze. Datensätze können während des Ladevorgangs in einer Coupling-Facility-Datentabelle nur dann durchsucht werden, falls der angeforderte Schlüssel im Bereich derjenigen Datensätze zu finden ist, die bereits geladen worden sind.

Die Antwort LOADING kann auch für eine Coupling-Facility-Datentabelle zurückgegeben werden, deren Ladevorgang fehlgeschlagen ist. Weitere Informationen zu den Folgen, die sich beim Fehlschlagen des Ladevorgangs für eine Coupling-Facility-Datentabelle ergeben, können Sie der Beschreibung für den globalen Benutzerexit XD TLC unter [Exits zur Datentabellenverwaltung \(XDTRD, XD TAD und XD TLC\)](#) entnehmen.

Falls Ihr Anwendungsprogramm die Bedingung LOADING ständig oder zu häufig feststellt, vergewissern Sie sich, dass dies nicht durch kollidierende FILE-Definitionen verursacht wird, die dieselbe Datei referenzieren.

100 LOCKED

RESP2-Werte:

106

In der Leseanforderung war das Schlüsselwort UPDATE bzw. eines der Schlüsselwörter CONSISTENT oder REPEATABLE für die Leseintegrität angegeben oder die FILE-Ressourcendefinition gibt die Leseintegrität an. VSAM hält jedoch eine beibehaltene Sperre für den Datensatz (siehe [„Beibehaltene und aktive Sperren“](#) auf Seite 389).

Der Schlüssel des gesperrten Datensatzes wird nicht an Ihr Anwendungsprogramm zurückgegeben. Falls Sie diese Bedingung behandeln und die Steuerung an Ihr Programm zurückgegeben wird, kann der Suchvorgang fortgesetzt werden und den Datensatz abrufen, der auf den gesperrten Datensatz folgt, indem eine weitere Anforderung READNEXT abgesetzt wird.

Die Bedingung LOCKED kann auch bei einer Anforderung für eine wiederherstellbare Coupling-Facility-Datentabelle auftreten, die das Sperrenmodell verwendet, falls der gelesene Datensatz durch eine beibehaltene Sperre gesperrt ist. Informationen zum Untersuchen von beibehaltenen Sperren für Datensätze in einer Coupling-Facility-Datentabelle finden Sie unter [Beibehaltene Sperren der Coupling-Facility-Datentabelle](#).

Standardaktion: Die Task wird mit dem Code AEX8 abnormal beendet.

70 NOTAUTH

RESP2-Werte:

101

Eine Ressourcensicherheitsprüfung ist für FILE(dateiname) fehlgeschlagen.

Standardaktion: Die Task wird abnormal beendet.

13 NOTFND

RESP2-Werte:

80

Der Versuch, einen Datensatz anhand des angegebenen Sucharguments abzurufen, war erfolglos. Dies kann auftreten, falls der Befehl READNEXT unmittelbar auf einen Befehl STARTBR folgt, der den Schlüssel des letzten Datensatzes in der Datei angegeben hat (ein vollständiger Schlüssel von Werten X'FF').

81

Die Option XRBA war angegeben und der Wert von RIDFLD war größer als 4 GB, bei der Datei handelt es sich jedoch nicht um eine ESDS-Datei mit erweiterter Adressierung.

Standardaktion: Die Task wird abnormal beendet.

101 RECORDBUSY

RESP2-Werte:

107

Die Leseintegrität NOSUSPEND ist in der Anforderung angegeben, aber VSAM hält eine aktive Sperre für den Datensatz, was einen Wartestatus für die Anforderung zur Folge hätte (siehe [„Beibehaltene und aktive Sperren“](#) auf Seite 389).

Der Schlüssel des gesperrten Datensatzes wird nicht an Ihr Anwendungsprogramm zurückgegeben. Falls Sie diese Bedingung behandeln und die Steuerung an Ihr Programm zurückgegeben wird, kann der Suchvorgang fortgesetzt werden und den Datensatz abrufen, der auf den gesperrten Datensatz folgt, indem eine weitere Anforderung READNEXT abgesetzt wird.

Standardaktion: Die Task wird mit dem Code AEX9 beendet.

53 SYSIDERR

RESP2-Werte:

130

Die Option SYSID gibt einen Namen an, bei dem es sich weder um das lokale System noch um ein fernes System handelt, das durch eine CONNECTION- oder IPCONN-Definition definiert ist. Die Bedingung SYSIDERR tritt außerdem auf, wenn die Verbindung zu dem fernen System bekannt, aber nicht verfügbar ist. Bei einer IPCONN-Definition tritt die Bedingung SYSIDERR auf, falls die Verbindung bekannt ist, jedoch entweder das lokale oder das ferne System keine Dateisteuerungsbefehle unterstützen, die mit IP-Interkonnektivität per Funktionsverlagerung übertragen werden.

131

Für eine Coupling-Facility-Datentabelle ist die Verbindung zum Server für Coupling-Facility-Datentabellen fehlgeschlagen. Dies könnte dadurch verursacht worden sein, dass der Server selbst fehlgeschlagen ist, oder dass der Server zwar verfügbar ist, CICS jedoch keine Verbindung zu ihm herstellen konnte.

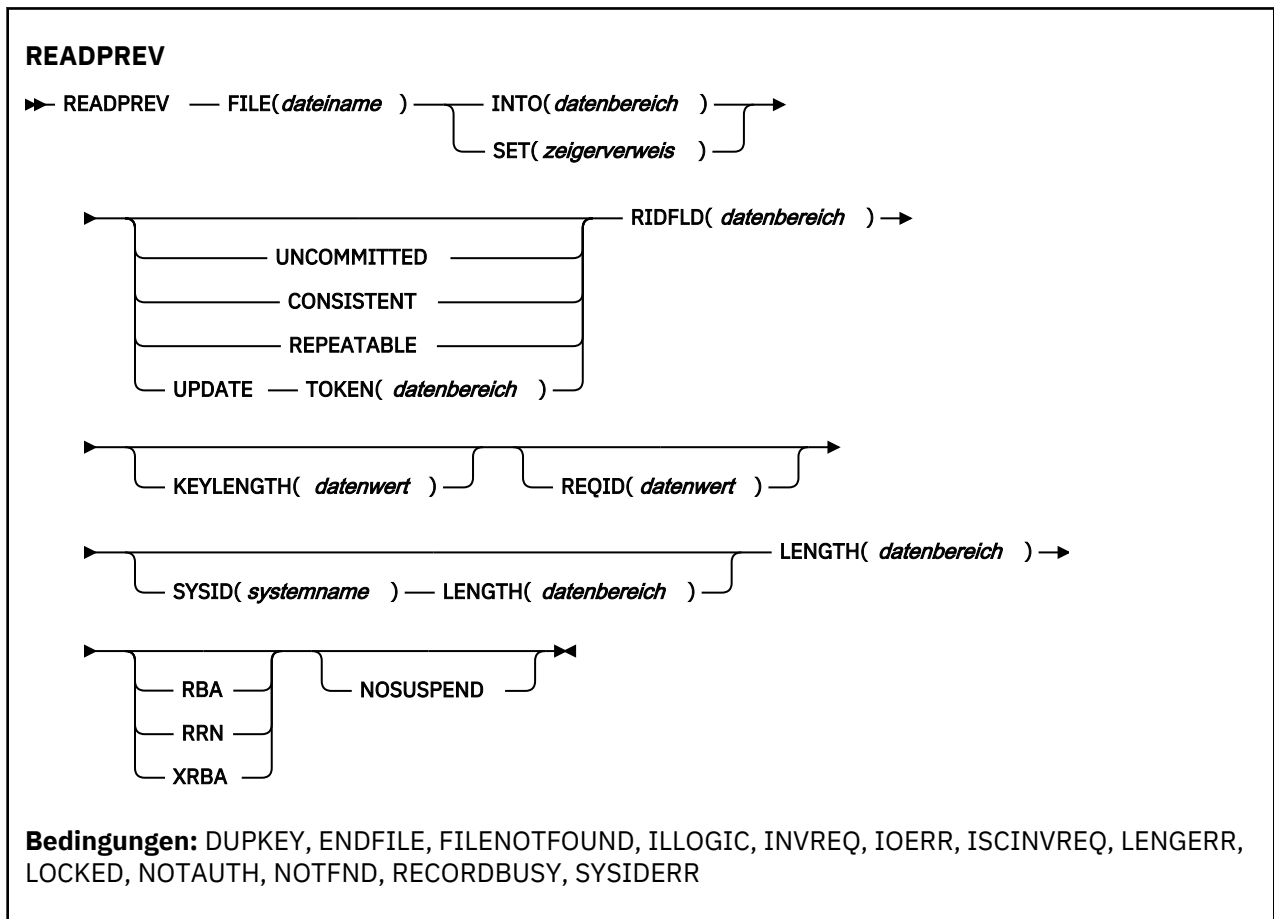
132

Der Befehl READNEXT wurde für eine Coupling-Facility-Datentabelle ausgegeben, die nicht mehr vorhanden ist. Vermutlich liegt dies an einem Fehler der Coupling-Facility, durch den der Server für Coupling-Facility-Datentabellen ebenfalls fehlschlägt. Informationen zum erneuten Starten eines Servers für Coupling-Facility-Datentabellen und zum erneuten Laden einer Tabelle finden Sie unter [Server für Coupling-Facility-Datentabellen einrichten und ausführen](#).

Standardaktion: Die Task wird abnormal beendet.

READPREV

Liest beim Durchsuchen einer Datei den vorherigen Datensatz. Gilt nur für VSAM und Datentabellen.



Dieser Befehl ist threadsicher, wenn für die Datei, auf die er sich bezieht, Folgendes zutrifft:

- Sie ist als fern definiert und der Befehl wird über eine IPIC-Verbindung durch Funktionsverlagerung an eine ferne CICS-Region übertragen.
- Sie ist als lokale VSAM-Datei, als VSAM-RLS-Datei oder als Coupling-Facility-Datentabelle definiert.

Dieser Befehl ist nicht threadsicher, wenn für die Datei, auf die er sich bezieht, Folgendes zutrifft:

- Sie ist als fern definiert und der Befehl wird über eine Nicht-IPIC-Verbindung durch Funktionsverlagerung übertragen.
- Sie ist als gemeinsam genutzte Datentabelle oder als BDAM-Datei definiert.

Beschreibung

Durch die wiederholte Verwendung des Befehls READPREV können Datensätze in umgekehrter Reihenfolge aus einer VSAM-Datei auf einem lokalen oder fernen System gelesen werden.

Eine solche Serie von sequenziellen Lesebefehlen wird als *Suchvorgang* in der Datei bezeichnet. Ein Suchvorgang kann auch eine Serie von Befehlen READNEXT und READPREV in beliebiger Reihenfolge umfassen. Ein Suchvorgang muss mit dem Befehl STARTBR initialisiert werden, um den Anfang des Suchvorgangs anzugeben; zur Beendigung muss der Befehl ENDBR verwendet werden.

In der Option RIDFLD müssen Sie einen Datenbereich angeben, der groß genug für eine vollständige Kennung von Datensätzen in der Datei ist, also für den vollständigen Schlüssel, die relative Byteadresse (RBA) oder die relative Satznummer (RRN). Dieser Datenbereich wird sowohl als Ausgabeparameter als auch als Eingabeparameter verwendet.

Der Datenbereich wird als Ausgabeparameter verwendet, wenn CICS beim Abschluss jedes Befehls READPREV die vollständige Kennung des soeben abgerufenen Satzes in den Datenbereich der Option RIDFLD stellt. CICS behält anschließend diese Kennung bei, um die Position zu kennzeichnen, an der der nächste Befehl READPREV fortgesetzt werden muss.

Der Datenbereich kann auch als Eingabeparameter verwendet werden. Eine Änderung des Wertes für RIDFLD vor einer Ausgabe des nächsten Befehls READPREV bewirkt, dass dieser Befehl den Suchvorgang an der neuen Kennung positioniert, von der aus er wie gewohnt fortgesetzt wird. Bei der geänderten Satznummer muss es sich immer um einen vollständigen Schlüssel, eine relative Byteadresse (RBA) oder eine relative Satznummer (RRN) handeln. Es kann weder ein generischer Schlüssel angegeben sein, noch kann ein Suchvorgang, der mit der Option GENERIC gestartet wurde, einen Befehl READPREV beinhalten.

Falls Sie einen Befehl READPREV unmittelbar nach einem Befehl STARTBR angeben, muss die Option RIDFLD des Befehls STARTBR den Schlüssel eines Datensatzes angeben, der in der Datei vorhanden ist; andernfalls tritt die Bedingung NOTFND auf.

Ein Befehl READPREV, der auf einen Befehl READNEXT bzw. einen Befehl STARTBR oder RESETBR folgt, bei dem kein 'letzter' Schlüsselwert angegeben war, wird so behandelt, als ob der Wert für RIDFLD geändert worden wäre, und führt (wie oben beschrieben) zu einer Neupositionierung.

Dateien durch Zugriff im RLS-Modus lesen

Für Dateien, auf die im RLS-Modus zugegriffen wird, können Sie das Schlüsselwort UPDATE in die Anforderung READPREV einbeziehen, um einige Datensätze beim Suchvorgang zu aktualisieren. Bei Angabe von UPDATE müssen Sie ebenfalls TOKEN angeben. Anschließend können Sie den Datensatz durch Ausgabe eines Befehls DELETE oder REWRITE aktualisieren, in dem der von der Suchfunktion zurückgegebene Wert für TOKEN angegeben ist.

Anmerkung: Die Verwendung von TOKEN ohne das Schlüsselwort UPDATE impliziert automatisch die Verwendung von UPDATE.

Für die Verwendung der Option UPDATE gelten die folgenden Regeln:

- Sie können die Option UPDATE bei einem Befehl READPREV nur dann angeben, wenn auf die Datei im RLS-Modus zugegriffen wird. Falls Sie die Option UPDATE für eine Datei angeben, auf die im Nicht-RLS-Modus zugegriffen wird, gibt CICS den Wert INVREQ zurück.
- Sie können die Option UPDATE bei einem Befehl READPREV, jedoch nicht bei einem Befehl STARTBR oder RESETBR angeben.
- Sie können Anforderungen mit der Option UPDATE und ohne Aktualisierung in demselben Suchvorgang kombinieren.
- CICS behält die Option UPDATE eines Befehls READPREV nicht automatisch für den nächsten Befehl bei.

CICS unterstützt in der Sequenz des Suchvorgangs nur einen einzigen Wert für TOKEN; in jedem Befehl READPREV macht der Wert für TOKEN den vorherigen Wert ungültig.

Sperren für Option UPDATE

Durch die Angabe der Option UPDATE für einen Befehl READPREV wird eine exklusive Sperre übernommen. Die Dauer dieser exklusiven Sperre innerhalb eines Suchvorgangs ist von der Aktion abhängig, die Ihr Anwendungsprogramm ausführt:

- Falls Sie den letzten Datensatz, der beim Suchvorgang durch einen Befehl READPREV mit der Option UPDATE unter Verwendung des zugehörigen Tokens angefordert wurde, mit DELETE oder REWRITE löschen bzw. neu schreiben wollen, bleibt die Sperre wie folgt aktiv:
 - Wenn die Datei wiederherstellbar ist, wird die Sperre nach Abschluss des nächsten Synchronisationspunktes oder Rollbacks freigegeben.
 - Wenn die Datei nicht wiederherstellbar ist, wird die Sperre spätestens beim Abschluss von ENDBR, jedoch möglicherweise auch früher freigegeben.
- Falls Sie den letzten gelesenen Datensatz *nicht* aktualisieren, gibt CICS die exklusive Sperre frei, sobald Ihr Programm entweder einen weiteren Befehl READNEXT bzw. READPREV ausgibt oder den Suchvorgang beendet.

Wichtig: Der Befehl UNLOCK bewirkt *keine* Freigabe einer exklusiven Sperre, die von VSAM für einen durch READPREV UPDATE angeforderten Datensatz gehalten wird. Ein Befehl UNLOCK beim Suchvorgang macht den Wert für TOKEN ungültig, der durch die letzte Anforderung zurückgegeben wurde.

Sperren für Leseintegrität

Die Angabe einer der Optionen für die Leseintegrität übernimmt eine gemeinsame Sperre bei jedem Befehl READPREV. Die Dauer dieser gemeinsamen Sperren bei einem Suchvorgang ist vom Typ der Leseintegrität abhängig, die Sie angeben:

- Bei Angabe der Leseintegrität CONSISTENT wird die gemeinsame Sperre nur für die Dauer jeder Leseanforderung gehalten, also bis zur Rückgabe des Datensatzes an Ihr Programm.
- Bei Angabe der Leseintegrität REPEATABLE wird die gemeinsame Sperre für die Dauer der Arbeitseinheit gehalten, in der der Suchvorgang ausgeführt wird. In diesem Fall könnte Ihr Programm viele gemeinsame Sperren anfordern, was die Erteilung von exklusiven Sperren für Aktualisierungsfunktionen verhindert. Die Leseintegrität REPEATABLE sollte bei einem Suchvorgang mit Vorsicht verwendet werden.

Funktionsverlagerung bei READPREV mit UPDATE oder Leseintegrität

Falls ein Befehl READPREV, der die Option UPDATE oder eine der Leseintegritätsoptionen angibt, per Funktionsverlagerung an ein Mitglied der CICS-Produktfamilie übertragen wird, das die Option UPDATE oder die Leseintegritätsoptionen nicht unterstützt, schlägt die Anforderung fehl:

- Bei Verwendung eines ISC-Links empfängt die Anforderung einen Abbruch mit dem Code ATNI.
- Bei Verwendung einer MRO-Verbindung empfängt die Anforderung einen AXF8-Abbruch.

AXF8 ist ein Abbruchcode, der von der Absenderseite einer Anforderung mit Funktionsverlagerung empfangen wird. Er gibt an, dass versucht wurde, eine Anforderung, in der eine Option UPDATE angegeben ist, über eine MRO-Verbindung an eine CICS-Region zu senden, die Aktualisierungs- oder Leseintegritätsoptionen nicht unterstützt.

Leseintegrität

Wenn im RLS-Modus auf eine Datei zugegriffen wird, können Leseanforderungen ohne Aktualisierungsoption die Leseintegritätsoptionen UNCOMMITTED, CONSISTENT oder REPEATABLE angeben.

Falls Sie keines dieser Schlüsselwörter angeben, verwendet CICS den im Parameter READINTEG der FILE-Ressourcendefinition angegebenen Wert, dessen Standardeinstellung UNCOMMITTED lautet.

Wenn Sie die Leseintegritätsstufe verwenden wollen, die im Schlüsselwort READINTEG der FILE-Definition angegeben ist, und anschließend von der Verwendung einer lokalen Datei zur Verwendung einer fernen Datei wechseln müssen oder die Position einer fernen Datei ändern, muss Folgendes sichergestellt sein:

- Die ferne File-Ownning Region (FOR) hat den Stand von CICS Transaction Server for OS/390 Version 1 Release 1 (oder höher).
- Die FILE-Definition auf dem fernen System gibt Folgendes an:

- RLS-Modus
- Korrekte Leseintegritätswerte für Ihre Anwendung

Beibehaltene und aktive Sperren

RECORDBUSY bezieht sich auf aktive Sperren, LOCKED bezieht sich auf beibehaltene Sperren.

Diese Sperren wirken sich auf Anforderungen READPREV aus, die Sperren anfordern, also auf Aktualisierungsanforderungen und Anforderungen mit Leseintegrität. Auf diese Arten von Anforderungen READPREV wird in den nächsten Listenelementen Bezug genommen. Andere Anforderungen READPREV werden durch beibehaltene oder aktive Sperren nicht beeinflusst.

- Anforderungen READPREV für Datensätze mit *beibehaltenen* Sperren werden immer mit einer Antwort LOCKED zurückgewiesen.
- Anforderungen READPREV für Datensätze mit *aktiven* Sperren warten auf die Freigabe der Sperre. Ausgenommen sind Fälle, bei denen das Schlüsselwort NOSUSPEND angegeben ist; hier gibt CICS die Antwort RECORDBUSY zurück.

Optionen

CONSISTENT (nur RLS)

Der Datensatz wird mit der Leseintegritätsstufe gelesen, die durch eine gemeinsame VSAM-Sperre bereitgestellt wird, die für die Dauer der Anforderung besteht.

Falls der Datensatz durch eine andere Task geändert wird, die daher eine exklusive Sperre hält, wartet die Anforderung READPREV (außer bei Angabe von NOSUSPEND) folgendermaßen auf den Abschluss der Aktualisierung:

- Bei einer Anforderung READPREV für eine nicht wiederherstellbare Datei wird der die Anforderung READPREV ausgeführt, sobald eine die Aktualisierung vornehmende VSAM-Anforderung abgeschlossen ist.
- Bei einer Anforderung READPREV für eine wiederherstellbare Datei wird die Anforderung READPREV ausgeführt, sobald die Aktualisierungstask ihren nächsten Synchronisationspunkt oder Rollback abschließt.

FILE(dateiname)

Gibt den Namen der zu durchsuchenden Datei an.

Bei Angabe eines Wertes für SYSID wird davon ausgegangen, dass sich die Datei, auf die sich dieser Dateiname bezieht, auf einem fernen System befindet, wobei es ohne Belang ist, ob der Name in der FILE-Ressourcendefinition definiert ist. Andernfalls wird mithilfe der FILE-Definition ermittelt, ob sich die Datei auf einem lokalen oder fernen System befindet.

INTO(datenbereich)

Gibt den Datenbereich an, in den der aus der Datei abgerufene Datensatz geschrieben werden soll.

KEYLENGTH(datenwert)

Gibt die Länge (Halbwort-Binärzahl) des Schlüssels an, der in der Option RIDFLD angegeben ist. Ausgenommen hiervon ist die Angabe der relativen Byteadresse (RBA) oder der relativen Satznummer (RRN); in diesem Fall ist die Option KEYLENGTH nicht gültig. Falls die angegebene Länge von der für die Datei definierten Länge abweicht, tritt die Bedingung INVREQ auf.

Für ferne Dateien kann der Wert von KEYLENGTH in der FILE-Definition angegeben werden. Falls die Option KEYLENGTH dort nicht definiert ist, nicht im Anwendungsprogramm angegeben ist und der Schlüssel länger als 4 Zeichen ist, lautet der Standardwert 4.

LENGTH(datenbereich)

Gibt die Länge des Datenbereichs, in den der Datensatz einzufügen ist, als Halbwort-Binärwert an. Beim Abschluss des Befehls READNEXT enthält der Parameter LENGTH die tatsächliche Länge des Datensatzes.

Diese Option muss angegeben werden, wenn SYSID angegeben wird.

Falls sich die Datei auf einem fernen System befindet, muss der Parameter LENGTH hier nicht festgelegt werden, aber in der FILE-Ressourcendefinition festgelegt sein.

Falls sich die Datei auf einem lokalen System befindet, muss der Parameter LENGTH mithilfe der Option INTO für Datensätze mit variabler Länge festgelegt werden, jedoch nicht für Datensätze mit fester Länge. Die Angabe der Länge für Datensätze mit fester Länge ist jedoch aus den folgenden Gründen ratsam:

- Sie bewirkt die Prüfung, ob der gelesene Datensatz nicht zu lang für den verfügbaren Datenbereich ist.
- Bei Suchvorgängen für Datensätze mit fester Länge in einem Bereich, der länger als die Datensätze ist, auf die zugegriffen wird, wird für COBOL-, C- und PL/I-Anwendungen sowie für Anwendungen in Assemblersprache die Bedingung LENGERR ausgelöst, falls die Option LENGTH angegeben ist. Wenn die Option LENGTH nicht explizit angegeben ist, wird die Bedingung LENGERR zurückgegeben, wenn die Länge aus dem Datenbereich abgeleitet werden kann. Überschreitet die angegebene Länge die Länge des Datensatzes, verwendet CICS für das Verschieben die größere Länge. Falls der Zielbereich im Anwendungsprogramm nicht groß genug ist, wird der Speicher über den Zielbereich hinaus überlagert.

Bei Angabe der Option SET müssen Sie nicht ebenfalls die Option LENGTH angeben.

Bei einem Suchvorgang für einen Zieldatenbereich, der länger als der gelesene Datensatz ist, ist der Inhalt des Zieldatenbereichs vom Ende des abgerufenen Datensatzes bis zum Ende des Zieldatenbereichs unvorhersehbar.

Falls Sie die Option INTO angeben, muss das Argument LENGTH ein Datenbereich sein, der den größten vom Programm akzeptierten Datensatz angibt. Falls der abgerufene Datensatz länger als der in der Option LENGTH angegebene Wert ist, wird der Datensatz auf den angegebenen Wert abgeschnitten und die Bedingung LENGERR ausgelöst. In diesem Fall wird der mit LENGTH angegebene Datenbereich auf die Länge des Datensatzes vor dem Abschneiden gesetzt.

Beachten Sie, dass ein Dateisteuerungsbefehl, der für einen Datensatz variabler Länge in einer Datei, die auf dem lokalen CICS-System definiert ist, mit der Bedingung LENGERR fehlschlägt, wenn die Länge nicht angegeben wird. Wird derselbe Befehl jedoch für eine Datei ausgegeben, die auf einem fernen System definiert ist, schlägt der Befehl nicht fehl.

NOSUSPEND (nur RLS)

Die Anforderung wartet nicht, wenn der Datensatz von VSAM mit einer aktiven Sperre gesperrt wurde. Dies gilt auch für Datensätze, die infolge eines Deadlocks gesperrt wurden.

Anmerkung: Anforderungen mit der Angabe NOSUSPEND warten mindestens eine Sekunde, bevor CICS die Antwort RECORDBUSY zurückgibt.

RBA

(Nur VSAM KSDS- oder ESDS- Basisdateien oder von CICS verwaltete Datentabellen, jedoch keine Pfade) Gibt an, dass das Datensatz-ID-Feld, das in der Option RIDFLD angegeben wurde, eine relative Byteadresse (RBA) enthält.

Diese Option muss angegeben sein, wenn die Option RBA im Befehl STARTBR oder RESETBR angegeben war. Sie darf nicht angegeben sein, wenn im Befehl STARTBR oder RESETBR die Option RBA nicht angegeben war.

Die Option RBA kann für Folgendes nicht verwendet werden:

- Benutzerverwaltete Datentabellen
- Coupling-Facility-Datentabellen
- Jede KSDS-Datei, die im RLS-Zugriffsmodus geöffnet ist
- Schlüsseladressierte Dateien mit erweiterter Adressierung

Von einer Verwendung der Option RBA wird darüber hinaus bei ESDS-Dateien mit einer Größe von mehr als 4 GB abgeraten. (Verwenden Sie stattdessen XRBA.)

REPEATABLE (nur RLS)

Der Datensatz wird mit der Leseintegritätsstufe gelesen, die durch eine gemeinsame VSAM-Sperre bereitgestellt wird, die für die Dauer der Arbeitseinheit (UOW) besteht, in der die Leseanforderung abgesetzt wird.

Falls der Datensatz durch eine andere Task geändert wird, die daher eine exklusive Sperre hält, wartet die Anforderung READPREV (außer bei Angabe von NOSUSPEND) folgendermaßen auf den Abschluss der Aktualisierung:

- Bei einer wiederherstellbaren Datei wird die Anforderung READPREV ausgeführt, sobald die aktualisierende Transaktion ihren nächsten Synchronisationspunkt oder Rollback abschließt.
- Bei einer nicht wiederherstellbaren Datei wird die Anforderung READPREV ausgeführt, sobald die VSAM-Anforderung, die die Aktualisierung vornimmt, abgeschlossen ist.

Nach Abschluss der Anforderung READPREV bleibt der Datensatz für die Task gesperrt, die die Anforderung READPREV ausgegeben hat. Andere Tasks können den Datensatz weiterhin lesen, aber der Datensatz kann von keiner anderen Task aktualisiert werden, bis die Task, die die Anforderung READPREV ausgegeben hat, ihren nächsten Lesevorgang oder Rollback ausführt.

REQID(datenwert)

Gibt in Form eines Halbwort-Binärwertes eine eindeutige Anforderungskennung für einen Suchvorgang an, mit dessen Hilfe Mehrfachsuchvorgänge für eine Datei gesteuert werden. Falls diese Option nicht angegeben ist, wird vom Standardwert Null ausgegangen.

RIDFLD(datenbereich)

Gibt das Feld für die Satzkennung an. Der Inhalt kann ein Schlüssel, eine relative Byteadresse oder eine relative Satznummer sein. Bei einer relativen Byteadresse oder einer relativen Satznummer muss dieses Feld das Format einer Vollwort-Binärzahl besitzen. Für eine relative Byteadresse kann der Wert von RIDFLD größer-gleich 0 sein. Für eine relative Satznummer kann der Wert von RIDFLD größer-gleich 1 sein.

Nach Abschluss des Befehls READPREV wird dieses Feld von CICS mit der vollständigen Kennung des abgerufenen Datensatzes aktualisiert.

RRN

(Schlüsseladressierte VSAM-Dateien) Gibt an, dass das in der Option RIDFLD angegebene Feld für die Satzkennung eine relative Satznummer enthält.

SET(zeigerverweis)

Gibt den Zeigerverweis an, der auf die Adresse des abgerufenen Datensatzes gesetzt werden soll.

Falls bei einer Anwendung in Assemblersprache die Bedingung DUPKEY auftritt, ist das angegebene Register nicht festgelegt worden, kann jedoch aus DFHEITP1 geladen werden.

Der Zeigerverweis bis zum nächsten Befehl READPREV oder READNEXT gültig, der die Option SET für denselben Suchvorgang (REQID) und dieselbe Datei angibt. Nach einem Befehl ENDBR oder SYNC-POINT ist der Zeiger nicht mehr gültig. Falls Sie die Daten in dem durch den Zeiger adressierten Feld beibehalten wollen, müssen Sie sie in Ihren eigenen Datenbereich verschieben.

Falls dem Anwendungsprogramm DATALOCATION(ANY) zugeordnet ist, kann die im SET-Zeiger zurückgegebene Adresse ober- oder unterhalb der 16-MB-Grenze liegen.

Falls dem Anwendungsprogramm DATALOCATION(BELOW) zugeordnet ist, liegt die vom SET-Zeiger zurückgegebene Adresse unterhalb der 16-MB-Grenze.

Falls für die aktive Task TASKDATAKEY(USER) angegeben ist, befinden sich die zurückgegebenen Daten im Benutzerschlüsselspeicher, andernfalls befinden sie sich im CICS-Schlüsselspeicher.

SYSID(systemname)

Gibt den Namen des Systems an, an das die Anforderung gerichtet ist.

Falls Sie die Option SYSID angeben und beide Optionen RBA und RRN weglassen, müssen Sie ebenfalls die Optionen LENGTH und KEYLENGTH angeben.

TOKEN(datenbereich) (nur RLS)

Gibt als Vollwort-Binärwert eine eindeutige Kennung für diese Anforderung READPREV UPDATE zurück. Dieser Ausgabewert wird durch die Dateisteuerung an die anfordernde Task zurückgegeben, damit er zur Zuordnung einer nachfolgenden Anforderung REWRITE oder DELETE (bzw. UNLOCK) mit dem für diesen Befehl READPREV zurückgegebenen Datensatz verwendet werden kann.

Ihr Anwendungsprogramm muss den für TOKEN zurückgegebenen Wert bei einem anschließenden Befehl REWRITE oder DELETE angeben, um den Datensatz zu kennzeichnen, der neu geschrieben oder gelöscht wird. Ihr Anwendungsprogramm kann auch den für die Option TOKEN zurückgegebenen Wert bei einem nachfolgenden Befehl UNLOCK angeben, um das Token zu kennzeichnen, das ungültig gemacht wird.

Sie müssen die Option TOKEN angeben, wenn Sie die Option UPDATE angeben (bei Angabe der Option TOKEN ohne die Option UPDATE wird jedoch UPDATE angenommen).

CICS unterstützt für einen bestimmten REQID-Wert jeweils nur einen einzigen aktiven Wert für TOKEN. Ein Wert für TOKEN bleibt daher nur bis zum nächsten Befehl READNEXT, READPREV oder ENDBR für diesen Suchvorgang bzw. bis zu einem Befehl REWRITE, DELETE oder UNLOCK gültig (je nachdem, was zuerst eintritt).

TOKEN kann durch Funktionsverlagerung übertragen werden. Falls jedoch eine Anforderung mit Angabe von TOKEN per Funktionsverlagerung an ein Mitglied der CICS-Produktfamilie übertragen wird, das dieses Schlüsselwort nicht erkennt, schlägt die Anforderung fehl.

UNCOMMITTED

Der Datensatz wird ohne Leseintegrität gelesen. Es werden die aktuellen Daten so zurückgegeben, wie sie für VSAM bekannt sind. Es wird nicht versucht, diese Leseanforderung mit einer etwaigen gleichzeitigen Aktualisierungsaktivität für denselben Datensatz zu serialisieren. Es besteht die Möglichkeit, dass der Datensatz gerade durch eine andere Transaktion aktualisiert wird, weshalb sich der Wert des Datensatzes später ändern kann, falls diese Aktualisierung anschließend zurückgesetzt wird.

UPDATE (nur RLS)

Gibt an, dass der Datensatz zur Aktualisierung oder Löschung abgerufen werden soll. Falls Sie sowohl diese Option als auch die Option TOKEN nicht angeben, wird ein Lesezugriff angenommen.

Bei Angabe der Option UPDATE müssen Sie auch die Option TOKEN angeben.

Die Option UPDATE ist nur für Dateien gültig, die in der lokalen Region definiert sind.

XRBA

Gibt an, dass das in der Option RIDFLD angegebene Feld für die Satzkennung eine erweiterte relative Byteadresse enthält. Diese Option sollte bei Suchvorgängen für Datensätze in einer ESDS-Datei mit erweiterter Adressierung verwendet werden.

Sie können die Option XRBA bei einem Befehl READPREV nur dann angeben, wenn im zugehörigen Befehl STARTBR oder RESETBR ebenfalls XRBA angegeben war.

Auf KSDS-Dateien kann durch XRBA nicht zugegriffen werden.

Bedingungen

15 DUPKEY

RESP2-Werte:

140

Ein Datensatz wird mithilfe eines Alternativindex mit dem Attribut NONUNIQUEKEY abgerufen und es ist ein weiterer Alternativindexdatensatz mit demselben Schlüssel vorhanden.

Falls bei einer Anwendung in Assemblersprache die Option SET verwendet wird, ist das angegebene Register nicht festgelegt worden, kann jedoch aus DFHEITP1 geladen werden.

Standardaktion: Die Task wird abnormal beendet.

20 ENDFILE

RESP2-Werte:

90

Während des Suchvorgangs wurde eine Dateiendebedingung festgestellt.

Standardaktion: Die Task wird abnormal beendet.

12 FILENOTFOUND

RESP2-Werte:

1

Ein in der Option FILE referenzierter Dateiname ist nicht für CICS definiert.

Standardaktion: Die Task wird abnormal beendet.

21 ILLOGIC

Der gegenwärtig ausgeführte Suchvorgang wird beendet, wenn diese Bedingung auftritt.

RESP2-Werte (VSAM):

110

Es ist ein VSAM-Fehler aufgetreten, der zu keiner der anderen CICS-Antwortkategorien gehört.

Weitere Informationen sind bei EIBRCODE im EXEC-Schnittstellenblock verfügbar; Details finden Sie unter [EIB-Felder](#).

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

RESP2-Werte:

20

Die FILE-Definition lässt keine Aktualisierungen zu.

24

Ein Befehl READPREV wurde für eine Datei ausgegeben, für die der vorherige Befehl STARTBR oder RESETBR die Option GENERIC enthielt.

26

Die Option KEYLENGTH wurde angegeben und die angegebene Schlüssellänge ist nicht mit der Länge identisch, die für die Datei definiert ist, auf die sich diese Dateiangabe bezieht.

37

Der Typ der Satzkennung (z. B. Schlüssel oder relative Byteadresse), der für den Zugriff auf eine Datei während des Suchvorgangs verwendet wird, wurde geändert. Die Angabe eines Adressierungstyps bei einem Befehl STARTBR und die eines anderen Typs bei einem Befehl READPREV ist nicht möglich.

39

Ein Befehl READPREV wurde für eine BDAM-Datei ausgegeben.

41

Der Wert für REQID bzw. SYSID oder der Dateiname stimmt nicht mit dem eines erfolgreichen Befehls STARTBR überein.

52

Die Leseintegrität CONSISTENT ist nicht zulässig, weil es sich nicht um eine VSAM-Datei handelt, auf die im RLS-Modus zugegriffen wird.

53

Die Leseintegrität REPEATABLE ist nicht zulässig, weil es sich nicht um eine VSAM-Datei handelt, auf die im RLS-Modus zugegriffen wird.

54

Die Option UPDATE ist nicht zulässig, weil es sich nicht um eine VSAM-Datei handelt, auf die im RLS-Modus zugegriffen wird.

55

Die Leseintegrität NOSUSPEND ist nicht zulässig, weil es sich nicht um eine VSAM-Datei handelt, auf die im RLS-Modus zugegriffen wird.

57

Ein Versuch, die Datei zu aktualisieren, ist fehlgeschlagen, weil die zugeordnete VSAM-Datei den Verfügbarkeitsstatus (AVAILABILITY) RREPL hat, der Aktualisierungsoperationen mit Ausnahme solcher von Replikationsprogrammen nicht zulässt.

Standardaktion: Die Task wird abnormal beendet.

17 IOERR

RESP2-Werte:

120

Während des Suchvorgangs wurde ein E/A-Fehler festgestellt. Ein E/A-Fehler ist ein nicht erwartetes Ereignis, das nicht durch eine CICS-Bedingung abgedeckt ist.

Bei VSAM-Dateien gibt die Bedingung IOERR für gewöhnlich einen Hardwarefehler an. Weitere Informationen sind im EXEC-Schnittstellenblock verfügbar; Details finden Sie unter [EIB-Felder](#).

Bei einer Coupling-Facility-Datentabelle gibt eine Bedingung IOERR an, dass eine falsche Antwort von einem Coupling-Facility-Zugriff zurückgegeben wurde.

Standardaktion: Die Task wird abnormal beendet.

54 ISCINVREQ

RESP2-Werte:

70

Das ferne System gibt einen Fehler an, der keiner bekannten Bedingung entspricht.

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

RESP2-Werte:

10

Für eine Datei mit Datensätzen variabler Länge wurde weder die Option LENGTH noch die Option SET angegeben.

11

Die Länge des mit angegebener Option INTO gelesenen Datensatzes überschreitet den Wert, der in der Option LENGTH angegeben ist; der Datensatz wird abgeschnitten und der in der Option LENGTH bereitgestellte Datenbereich wird auf die tatsächliche Länge des Datensatzes gesetzt.

13

Für eine Datei mit Datensätzen fester Länge wurde eine falsche Länge angegeben.

Standardaktion: Die Task wird abnormal beendet.

100 LOCKED

RESP2-Werte:

106

In der Leseanforderung war das Schlüsselwort UPDATE bzw. eines der Schlüsselwörter CONSISTENT oder REPEATABLE für die Leseintegrität angegeben oder die FILE-Ressourcendefinition gibt die Leseintegrität an. VSAM hält jedoch eine beibehaltene Sperre für den Datensatz (siehe [„Beibehaltene und aktive Sperren“](#) auf Seite 401).

Der Schlüssel des gesperrten Datensatzes wird nicht an Ihr Anwendungsprogramm zurückgegeben. Falls Sie diese Bedingung behandeln und die Steuerung an Ihr Programm zurückgegeben wird, kann der Suchvorgang fortgesetzt werden und den Datensatz abrufen, der auf den gesperrten Datensatz folgt, indem eine weitere Anforderung READPREV abgesetzt wird.

Die Bedingung LOCKED kann auch bei einer Anforderung für eine wiederherstellbare Coupling-Facility-Datentabelle auftreten, die das Sperrenmodell verwendet, falls der gelesene Datensatz durch eine beibehaltene Sperre gesperrt ist. Informationen zum Untersuchen von beibehaltenen Sperren für Datensätze in einer Coupling-Facility-Datentabelle finden Sie unter [Beibehaltene Sperren der Coupling-Facility-Datentabelle](#).

Standardaktion: Die Task wird mit dem Code AEX8 abnormal beendet.

70 NOTAUTH

RESP2-Werte:

101

Eine Ressourcensicherheitsprüfung ist für FILE(dateiname) fehlgeschlagen.

Standardaktion: Die Task wird abnormal beendet.

13 NOTFND

RESP2-Werte:

80

Der Versuch, einen Datensatz anhand des angegebenen Sucharguments abzurufen, war erfolglos. Diese Bedingung kann unter anderem auftreten, wenn der Befehl READPREV unmittelbar auf einen Befehl STARTBR oder RESETBR folgt, in dem die Option GTEQ und der Schlüssel eines Datensatzes angegeben war, der nicht in der Datei vorhanden ist.

81

Die Option XRBA war angegeben und der Wert von RIDFLD war größer als 4 GB, bei der Datei handelt es sich jedoch nicht um eine ESDS-Datei mit erweiterter Adressierung.

Standardaktion: Die Task wird abnormal beendet.

101 RECORDBUSY

RESP2-Werte:

107

Die Leseintegrität NOSUSPEND ist in der Anforderung angegeben, aber VSAM hält eine aktive Sperre für den Datensatz, was einen Wartestatus für die Anforderung zur Folge hätte (siehe [„Beibehaltene und aktive Sperren“](#) auf Seite 401).

Der Schlüssel des gesperrten Datensatzes wird nicht an Ihr Anwendungsprogramm zurückgegeben. Falls Sie diese Bedingung behandeln und die Steuerung an Ihr Programm zurückgegeben wird, kann der Suchvorgang fortgesetzt werden und den Datensatz abrufen, der auf den gesperrten Datensatz folgt, indem eine weitere Anforderung READPREV abgesetzt wird.

Standardaktion: Die Task wird mit dem Code AEX9 abnormal beendet.

53 SYSIDERR

RESP2-Werte:

130

Die Option SYSID gibt einen Namen an, bei dem es sich weder um das lokale System noch um ein fernes System handelt, das durch eine CONNECTION- oder IPCONN-Definition definiert ist. Die Bedingung SYSIDERR tritt außerdem auf, wenn die Verbindung zu dem fernen System bekannt, aber nicht verfügbar ist. Bei einer IPCONN-Definition tritt die Bedingung SYSIDERR auf, falls die Verbindung bekannt ist, jedoch entweder das lokale oder das ferne System keine Dateisteuerungsbefehle unterstützen, die mit IP-Interkonnektivität per Funktionsverlagerung übertragen werden.

131

Für eine Coupling-Facility-Datentabelle ist die Verbindung zum Server für Coupling-Facility-Datentabellen fehlgeschlagen. Dies könnte dadurch verursacht worden sein, dass der Server selbst fehlgeschlagen ist, oder dass der Server zwar verfügbar ist, CICS jedoch keine Verbindung zu ihm herstellen konnte.

132

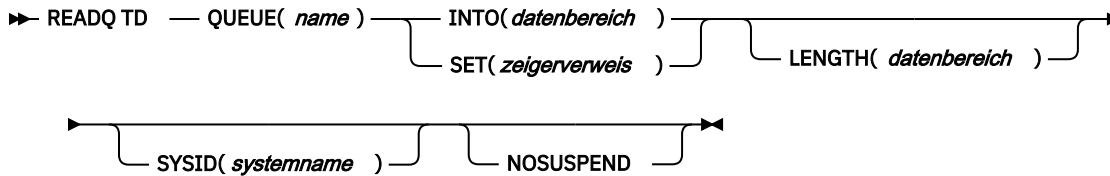
Der Befehl READPREV wurde für eine Coupling-Facility-Datentabelle ausgegeben, die nicht mehr vorhanden ist. Vermutlich liegt dies an einem Fehler der Coupling-Facility, durch den der Server für Coupling-Facility-Datentabellen ebenfalls fehlschlägt. Informationen zum erneuten Starten eines Servers für Coupling-Facility-Datentabellen und zum erneuten Laden einer Tabelle finden Sie unter [Server für Coupling-Facility-Datentabellen einrichten und ausführen](#).

Standardaktion: Die Task wird abnormal beendet.

READQ TD

Liest Daten aus der Warteschlange mit transienten Daten.

READQ TD



Bedingungen: DISABLED, INVREQ, IOERR, ISCINVREQ, LENGERR, LOCKED, NOTAUTH, NOTOPEN, QBUSY, QIDERR, QZERO, SYSIDERR

Dieser Befehl ist threadsicher, wenn er für eine Warteschlange in einer lokalen CICS-Region verwendet oder mit der Funktionsverlagerung über eine IPIC-Verbindung in eine ferne CICS-Region übertragen wird. Er ist nicht threadsicher, wenn er mit der Funktionsverlagerung über eine andere Art von Verbindung in eine ferne CICS-Region übertragen wird.

Beschreibung

Der Befehl **READQ TD** liest transiente Daten aus einer Warteschlange (danach ist der Datensatz nicht mehr verfügbar).

Falls Sie die automatische Transaktionsinitialisierung (ATI) verwenden, sollte Ihre Bedingung testen, ob die Bedingung QZERO vorliegt, damit sichergestellt ist, dass die Beendigung einer automatisch initialisierten Task erst dann stattfindet, wenn die Warteschlange leer ist. Eine Einführung in ATI finden Sie unter [Automatische Transaktionsinitialisierung \(ATI\)](#).

Falls der Befehl **READQ TD** versucht, auf einen Datensatz in einer logisch wiederherstellbaren partitions-internen Warteschlange zuzugreifen, die gegenwärtig durch eine andere Task mit Daten beschrieben oder gelöscht wird, und es keine weiteren festgeschriebenen Datensätze gibt, wartet der Befehl, bis die Warteschlange nicht mehr für Ausgabe verwendet wird. Wurde jedoch die Option NOSUSPEND angegeben oder gibt es einen aktiven Befehl HANDLE CONDITION für QBUSY, tritt die Bedingung QBUSY auf.

Optionen

INTO(datenbereich)

Gibt den Benutzerdatenbereich an, in den die Daten gestellt werden sollen, die aus der Warteschlange mit transienten Daten gelesen werden.

LENGTH(datenbereich)

Gibt als Halbwort-Binärwert die Länge des zu lesenden Datensatzes an.

Falls Sie die Option INTO angeben, gibt das Argument LENGTH die maximale Länge der Daten an, die das Programm akzeptiert. Falls der angegebene Wert kleiner als Null ist, tritt eine Bedingung LENGERR auf und die Warteschlange wird nicht gelesen. Ist der Wert Null angegeben, tritt eine Bedingung LENGERR auf und die Warteschlange wird gelesen. Falls die Länge der Daten den angegebenen Wert überschreitet, werden die Daten auf diesen Wert abgeschnitten und die Bedingung LENGERR tritt auf. Nach Abschluss der Abrufoperation wird der Datenbereich auf die ursprüngliche Länge des Datensatzes gesetzt, der aus der Warteschlange gelesen wurde.

Falls Sie die Option INTO angeben, muss das Argument LENGTH nicht unbedingt angegeben werden, wenn die Länge vom Compiler aus der Variablen INTO generiert werden kann. Weitere Informationen zu Fällen, in denen das Argument LENGTH angegeben werden muss, finden Sie unter [„LENGTH-Optionen in CICS-Befehlen“](#) auf Seite 10.

NOSUSPEND

Gibt an, dass die Task nicht ausgesetzt wird, bis die Warteschlange verfügbar wird, falls das Anwendungsprogramm versucht, Daten aus einer Warteschlange zu lesen, die bereits für die Ausgabe verwendet wird. In einem solchen Fall tritt stattdessen die Bedingung QBUSY auf.

Falls ein Befehl **HANDLE CONDITION** für **QBUSY** aktiv ist, wenn der Befehl ausgeführt wird, ist allerdings zu beachten, dass dies auch die Standardaktion außer Kraft setzt und die Steuerung an die Benutzerkennung übergeben wird, die im Befehl **HANDLE CONDITION** angegeben ist. Dies hat Vorrang vor der Option **NOSUSPEND**, wird jedoch entweder durch **NOHANDLE** oder durch **RESP** natürlich zurückgewiesen.

Diese Option gilt nur für partitionsinterne Warteschlangen.

QUEUE(name)

Gibt den symbolischen Namen (1 bis 4 alphanumerische Zeichen) der Warteschlange an, aus der Daten gelesen werden sollen. Die benannte Warteschlange muss für CICS definiert worden sein.

Bei Angabe eines Wertes für **SYSID** wird davon ausgegangen, dass sich die Warteschlange auf einem fernen System befindet, wobei es ohne Belang ist, ob sie als ferne Warteschlange definiert ist oder nicht. Andernfalls wird mithilfe der Definition für die Warteschlange mit transienten Daten ermittelt, ob sich die Datei auf einem lokalen oder fernen System befindet.

SET(zeigerverweis)

Gibt den Zeigerverweis an, der auf die Adresse der aus der Warteschlange gelesenen Daten gesetzt werden soll. CICS fordert einen Bereich an, der groß genug für den Datensatz ist, und legt den Zeigerverweis auf die Adresse dieses Bereichs fest. Der Bereich wird beibehalten, bis ein anderer Befehl für transiente Daten ausgeführt wird. Sofern der Zeigerverweis nicht durch andere Befehle oder Anweisungen geändert wird, bleibt er bis zum nächsten Befehl **READQ TD** oder bis zum Ende der Task gültig.

Wenn dem Anwendungsprogramm **DATALOCATION(ANY)** zugeordnet ist, kann die Adresse der Daten ober- oder unterhalb der 16-MB-Grenze liegen.

Falls dem Anwendungsprogramm **DATALOCATION(BELOW)** zugeordnet ist und sich die Daten oberhalb der 16-MB-Grenze befinden, werden die Daten an eine Adresse unterhalb der 16-MB-Grenze kopiert und die Adresse dieser Kopie wird zurückgegeben.

Wenn für die aktive Task **TASKDATAKEY(USER)** angegeben wird und der Speicherschutz aktiv ist, befinden sich die zurückgegebenen Daten in einem Benutzerschlüssel. Wenn für die aktive Task **TASKDATAKEY(CICS)** angegeben wird und der Speicherschutz aktiv ist, befinden sich die zurückgegebenen Daten in einem CICS-Schlüssel.

SYSID(systemname)

(Nur ferne Systeme) Gibt den ein bis vier Zeichen langen Namen des Systems an, an das die Anforderung gerichtet ist.

Bedingungen

84 DISABLED

Tritt auf, wenn die Warteschlange inaktiviert wurde.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

Tritt auf, falls der Befehl **READQ** eine partitionsübergreifende Warteschlange benennt, die für die Ausgabe geöffnet wurde. Bei partitionsinternen Warteschlangen kann diese Bedingung nicht auftreten.

Standardaktion: Die Task wird abnormal beendet.

17 IOERR

Tritt auf, wenn es einen Ein-/Ausgabefehler gibt und der fehlerhafte Datensatz übersprungen wird.

Diese Bedingung tritt auf, wenn die Warteschlange gelesen werden kann. Wenn die Warteschlange nicht gelesen werden kann, tritt in einer solchen Situation die Bedingung **QZERO** auf.

Diese Bedingung tritt ebenfalls auf, wenn der Operand **FREE=CLOSE** in der Dateidefinition für eine Warteschlange verwendet sowie die Warteschlange geschlossen und erneut geöffnet wurde.

Standardaktion: Die Task wird abnormal beendet.

54 ISCINVREQ

Tritt auf, wenn das ferne System einen Fehler angegeben hat, der keiner bekannten Bedingung entspricht.

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

Tritt auf, wenn der Befehl **READQ** für INTO einen Bereich benennt, der nicht alle an die Anwendung zurückgegebenen Daten aufnehmen kann, oder wenn eine ungültige Länge angegeben wurde. Die Prüfung findet statt, nachdem der Exit XTDIN aufgerufen wurde.

Standardaktion: Die Task wird abnormal beendet.

100 LOCKED

Tritt auf, wenn die Anforderung nicht ausgeführt werden kann, weil die Nutzung der Warteschlange aufgrund des unbestätigten Fehlschlagens einer Arbeitseinheit eingeschränkt wurde. Dies kann bei einer beliebigen Anforderung für eine logisch wiederherstellbare Warteschlange stattfinden, die in der TDQUEUE-Ressourcendefinition mit WAIT(YES) und WAITACTION(REJECT) definiert ist.

Geben Sie WAIT(YES) und WAITACTION(QUEUE) in der TDQUEUE-Ressourcendefinition an, wenn die Transaktion warten soll.

Standardaktion: Die Task wird abnormal beendet.

70 NOTAUTH

Tritt auf, wenn eine Ressourcensicherheitsprüfung für QUEUE(name) fehlgeschlagen ist.

Standardaktion: Die Task wird abnormal beendet.

19 NOTOPEN

Tritt auf, wenn die Zieladresse geschlossen ist. Diese Bedingung gilt nur für partitionsübergreifende Warteschlangen.

Standardaktion: Die Task wird abnormal beendet.

25 QBUSY

Tritt auf, wenn ein Befehl **READQ TD** versucht, auf einen Datensatz in einer logisch wiederherstellbaren partitionsinternen Warteschlange zuzugreifen, die gegenwärtig durch eine andere Task mit Daten beschrieben oder gelöscht wird, und es keine weiteren festgeschriebenen Datensätze gibt.

Damit diese Bedingung auftritt, muss die Option NOSUSPEND angegeben oder ein Befehl HANDLE für die Bedingung aktiv sein.

Diese Option gilt nur für partitionsinterne Warteschlangen.

Standardaktion: Die Bedingung wird ignoriert.

44 QIDERR

Tritt auf, wenn das symbolische Ziel, das beim Befehl **READQ TD** verwendet werden soll, nicht zu finden ist.

Standardaktion: Die Task wird abnormal beendet.

23 QZERO

Tritt auf, wenn das Ziel (also die Warteschlange) leer ist oder das Ende der Warteschlange erreicht wurde.

Standardaktion: Die Task wird abnormal beendet.

53 SYSIDERR

Tritt auf, wenn die Option SYSID einen Namen angibt, bei dem es sich weder um das lokale System noch um ein fernes System handelt (das CICS durch die Definition einer CONNECTION- oder IPCONN-Ressource bekanntgegeben wird). Die Bedingung SYSIDERR tritt außerdem auf, wenn die Verbindung zu dem fernen System geschlossen ist.

Standardaktion: Die Task wird abnormal beendet.

Beispiele

Das folgende Beispiel zeigt, wie ein Datensatz aus einer partitionsinternen Datei (Warteschlange), bei der es sich hier um das Nachrichtenprotokoll des Steuersystems (Control System Message Log, CSML) handelt, in einen Datenbereich gelesen wird, der in der Anforderung angegeben ist:

```
EXEC CICS READQ TD  
  QUEUE('CSML')  
  INTO(DATA)  
  LENGTH(LDATA)
```

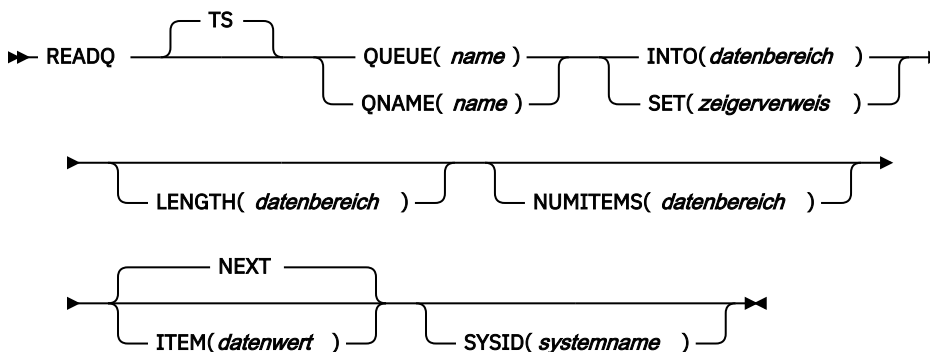
Das folgende Beispiel zeigt, wie ein Datensatz aus einer partitionsübergreifenden Datei (Warteschlange) mit Datensätzen fester Länge in einen durch CICS bereitgestellten Datenbereich gelesen wird. Der in der Option SET angegebene Zeigerverweis ist auf die Adresse des für den Datensatz reservierten Speicherbereichs gesetzt. Es wird angenommen, dass die Satzlänge bekannt ist.

```
EXEC CICS READQ TD  
  QUEUE(EX1)  
  SET(PREF)
```

READQ TS

Liest Daten aus einer Warteschlange für temporären Speicher.

READQ TS



Bedingungen: INVREQ, IOERR, ISCINVREQ, ITEMERR, LENGERR, NOTAUTH, QIDERR, SYSIDERR

Dieser Befehl ist threadsicher, wenn er für eine Warteschlange im Hauptspeicher oder im Zusatzspeicher entweder in einer lokalen CICS-Region verwendet oder mit der Funktionsverlagerung über eine IPIC-Verbindung in eine ferne CICS-Region übertragen wird. Er ist außerdem threadsicher, wenn er mit einer Warteschlange in einem gemeinsam genutzten temporären Speicherpool in einer z/OS-Coupling-Facility verwendet wird, die durch einen Server für gemeinsame Datennutzung mit temporären Speicher (TS-Server) verwendet wird. Der Befehl ist nicht threadsicher, wenn er mit der Funktionsverlagerung über einen anderen Verbindungstyp als IPIC in eine ferne CICS-Region übertragen wird.

Hinweis für dynamisches Transaktionsrouting: Die Verwendung dieses Befehls kann transaktionsübergreifende Affinitäten erzeugen, die die Nutzung des dynamischen Transaktionsroutings beeinträchtigen. Weitere Informationen zu Transaktionsaffinitäten finden Sie unter [Affinität](#).

Beschreibung

Der Befehl READQ TS ruft Daten aus einer Warteschlange für temporären Speicher im Haupt- oder Zusatzspeicher ab.

Optionen

INTO(*datenbereich*)

Gibt den Datenbereich an, in den die Daten geschrieben werden sollen. Der Datenbereich kann eine beliebige Variable, Struktur oder ein beliebiges Array sein.

ITEM(*datenwert*)

Gibt als Halbwort-Binärwert die Elementnummer des logischen Datensatzes an, der aus der Warteschlange abgerufen werden soll.

LENGTH(*datenbereich*)

Gibt als Halbwort-Binärwert die Länge des zu lesenden Datensatzes an.

Falls Sie die Option INTO angeben, muss das Argument LENGTH nicht unbedingt angegeben werden, wenn die Länge vom Compiler aus der Variablen INTO generiert werden kann.

Falls Sie die Option INTO angeben, definiert das Argument LENGTH die maximale Länge der Daten, die das Programm akzeptiert. Falls der angegebene Wert kleiner als Null ist, wird Null angenommen. Falls die Länge der Daten den angegebenen Wert überschreitet, werden die Daten auf diesen Wert abgeschnitten und die Bedingung LENGERR tritt auf.

Nach Abschluss der Abrufoperation wird der Datenbereich auf die ursprüngliche Länge des Datensatzes gesetzt, der aus der Warteschlange gelesen wurde.

Bei Angabe der Option SET muss die Option LENGTH ebenfalls angegeben sein.

Weitere Informationen zu Fällen, in denen die Option LENGTH angegeben werden muss, finden Sie unter „LENGTH-Optionen in CICS-Befehlen“ auf Seite 10.

NEXT

Gibt den Abruf für den nächsten sequenziellen logischen Satz an, der auf den letzten durch eine beliebige Task abgerufenen Datensatz folgt, oder den ersten Datensatz, falls zuvor kein Datensatz abgerufen wurde.



Achtung: Es ist möglich, dass sich zwei Tasks überlappen, falls sie während des Suchvorgangs die Steuerung verlieren. Beispiel: Task 1 ruft die Elemente 1, 3 und 6 ab, während Task 2 die Elemente 2, 4 und 5 abrufen. Die Verwendung des Befehls **READQ TS** mit der Option NEXT aus einem threadsicheren Programm erhöht die Wahrscheinlichkeit von sich überlappenden Tasks, weil diese parallel in ihren eigenen Tasksteuerblöcken ausgeführt werden. Falls die Reihenfolge beim Abrufen von Elementen wichtig ist, fügen Sie Serialisierungslogik zur Anwendung hinzu, damit das Durchsuchen der Warteschlange durch einen Einzelthread stattfindet. Dies gilt insbesondere dann, wenn die Anwendung mit der Option CONCURRENCY (THREADSAFE) oder CONCURRENCY (REQUIRED) definiert sein soll.

NUMITEMS(*datenbereich*)

Gibt ein Feld für eine Halbwort-Binärzahl an, in dem CICS eine Zahl speichert, die angibt, wie viele Elemente sich in der Warteschlange befinden. Dies findet nur bei einem normalen Abschluss des Befehls statt.

QNAME(*name*)

Als Alternative zu QUEUE gibt die Option QNAME den symbolischen Namen (1 - 16 Zeichen) der Warteschlange an, aus der Daten gelesen werden sollen. Falls der Name kürzer als 16 Zeichen ist, müssen Sie trotzdem ein 16 Zeichen langes Feld verwenden, das bei Bedarf mit Leerzeichen aufgefüllt wird.

QUEUE(*name*)

Gibt den symbolischen Namen (1 bis 8 Zeichen) der Warteschlange an, aus der Daten gelesen werden sollen. Falls der Name kürzer als 8 Zeichen ist, müssen Sie trotzdem ein 8 Zeichen langes Feld verwenden, das bei Bedarf mit Leerzeichen aufgefüllt wird.

SET(*zeigerverweis*)

Gibt den Zeigerverweis an, der auf die Adresse der abgerufenen Daten gesetzt wird. Sofern der Zeigerverweis nicht durch andere Befehle oder Anweisungen geändert wird, bleibt er bis zum nächsten Befehl READQ TS oder bis zum Ende der Task gültig.

Falls das Anwendungsprogramm mit DATALOCATION(ANY) definiert ist, kann die Adresse der Daten ober- oder unterhalb der 16-MB-Grenze liegen. Falls das Anwendungsprogramm mit DATALOCATION(BELOW) definiert ist, liegt die Adresse der Daten unterhalb der 16-MB-Grenze.

Wenn für die aktive Task TASKDATAKEY(USER) angegeben wird und der Speicherschutz aktiv ist, werden die Daten in einem Benutzerschlüssel zurückgegeben. Wenn für die aktive Task TASKDATAKEY(CICS) angegeben wird und der Speicherschutz aktiv ist, befinden sich die zurückgegebenen Daten in einem CICS-Schlüssel.

SYSID(systemname)

(Nur ferne und gemeinsam genutzte Warteschlangen) Gibt den ein bis vier Zeichen langen Systemnamen für das ferne System oder den Pool von gemeinsam genutzten Warteschlangen an, an das/den die Anforderung gerichtet ist. TSMODEL-Ressourcendefinitionen unterstützen nicht die Angabe eines SYSID-Wertes für eine Warteschlange, die sich in einem Pool für die gemeinsame Datennutzung von temporärem Speicher befindet. Verwenden Sie stattdessen die Option QUEUE oder QNAME. Die Verwendung eines expliziten SYSID-Wertes für einen Pool mit gemeinsam genutzten Warteschlangen macht die Unterstützung einer Tabelle für temporären Speicher (TST) erforderlich.

Bedingungen

16 INVREQ

Tritt in einer der folgenden Situationen auf:

- Die Warteschlange wurde durch internen Code von CICS erstellt.
- Der angegebene Warteschlangenname besteht ausschließlich aus binären Nullen.

Standardaktion: Die Task wird abnormal beendet.

17 IOERR

RESP2-Werte:

5

Für eine gemeinsam genutzte Warteschlange trat ein nicht wiederherstellbarer Ein-/Ausgabefehler auf.

Standardaktion: Die Task wird abnormal beendet.

54 ISCINVREQ

Tritt auf, wenn das ferne System einen Fehler angegeben hat, der keiner bekannten Bedingung entspricht.

Standardaktion: Die Task wird abnormal beendet.

26 ITEMERR

Tritt in einer der folgenden Situationen auf:

- Die angegebene Elementnummer ist ungültig (liegt also außerhalb des Bereichs der Elementnummern, die in die Warteschlange geschrieben wurden).
- Es wurde versucht, mit der Option NEXT (Standardeinstellung) Daten über das Ende der Warteschlange hinaus zu lesen.

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

Tritt auf, wenn die Länge der gespeicherten Daten den durch die Option LENGTH angegebenen Wert überschreitet.

Diese Bedingung gilt nur für die Option INTO und kann bei der Option SET nicht auftreten.

Standardaktion: Die Task wird abnormal beendet.

70 NOTAUTH

RESP2-Werte:

101

Eine Ressourcensicherheitsprüfung ist für QUEUE(name) fehlgeschlagen.

Standardaktion: Die Task wird abnormal beendet.

44 QIDERR

Tritt auf, wenn die angegebene Warteschlange weder im Haupt- noch im Zusatzspeicher gefunden wird.

Standardaktion: Die Task wird abnormal beendet.

53 SYSIDERR

RESP2-Werte:

4

Tritt in einer der folgenden Situationen auf:

- Die Option SYSID gibt einen Namen an, bei dem es sich nicht um das lokale System oder um ein fernes System handelt (das CICS durch eine CONNECTION- oder IPCONN-Definition bekanntgegeben wird).
- Bei Verwendung der IPIC-Konnektivität handelt es sich beim lokalen und/oder beim fernen System nicht um eine Region von CICS TS 4.2 oder höher.
- Die Verbindung zum fernen System wurde geschlossen.
- Die CICS-Region, in der der Befehl für den temporären Speicher ausgeführt wird, kann keine Verbindung zum TS-Server herstellen, der den TS-Pool verwaltet, von dem die referenzierte Warteschlange für temporären Speicher unterstützt wird. Diese Situation kann beispielsweise vorliegen, wenn die CICS-Region nicht für den Zugriff auf den Server für temporären Speicher berechtigt ist.

Sie kann ebenfalls vorliegen, falls der Server für temporären Speicher nicht gestartet wurde oder weil der Server fehlgeschlagen ist (bzw. gestoppt wurde), während CICS weiterhin ausgeführt wird.

Standardaktion: Die Task wird abnormal beendet.

Beispiele

Das folgende Beispiel zeigt, wie der erste (oder einzige) Datensatz aus einer Warteschlange für temporären Speicher in einen Datenbereich gelesen wird, der in der Anforderung angegeben ist. Die Option LENGTH für den Datenbereich erhält als Wert die Länge des Datensatzes.

```
EXEC CICS READQ TS  
      ITEM(1)  
      QUEUE(UNIQUE)  
      INTO(DATA)  
      LENGTH(LDATA)
```

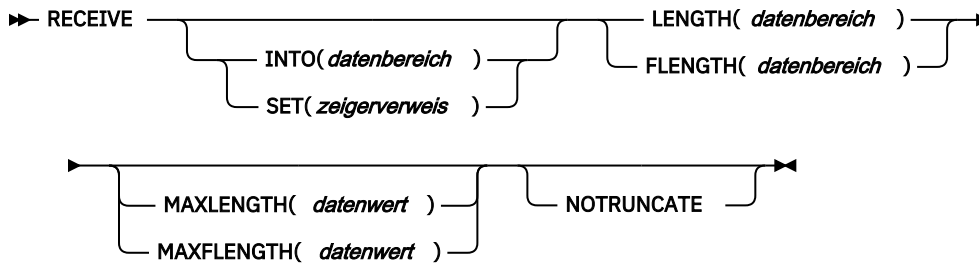
Das folgende Beispiel zeigt, wie der nächste Datensatz aus einer Warteschlange für temporären Speicher in einen Datenbereich gelesen wird, der von CICS bereitgestellt wird. Der Zeigerverweis, der durch die Option SET angegeben ist, wird auf die Adresse des für den Datensatz reservierten Speicherbereichs gesetzt; die Option LENGTH für den Datenbereich erhält als Wert die Länge des Datensatzes.

```
EXEC CICS READQ TS  
      QUEUE(DESCRQ )  
      SET(PREF)  
      LENGTH(LENG)  
      NEXT
```

RECEIVE (Standardformat für z/OS Communications Server)

Empfängt Daten von der CICS-Standardterminalunterstützung oder von einer Task, die nicht an ein Terminal angehängt ist.

RECEIVE (Standardformat)



Bedingungen: INVREQ, LENGERR, NOTALLOC

Beschreibung

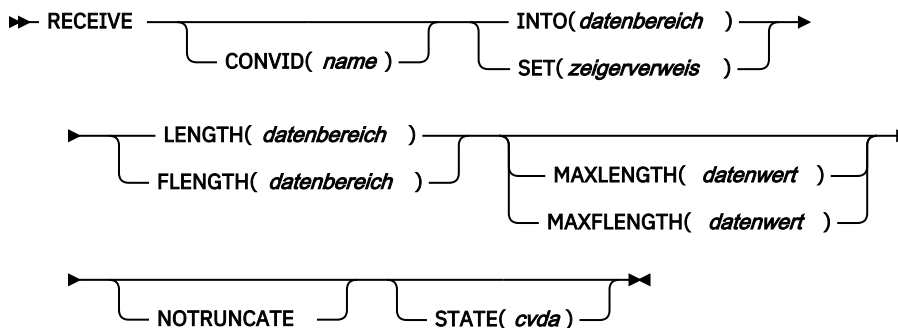
Diese Form des Befehls **RECEIVE** wird von allen von CICS unterstützten Terminals verwendet, für die die anderen **RECEIVE**-Beschreibungen nicht geeignet sind.

Falls Daten empfangen werden sollen, müssen Sie entweder die Option **INTO** oder die Option **SET** angeben. Wenn ein Befehl **RECEIVE** ausschließlich zu dem Zweck ausgegeben wird, ein AID-Zeichen zu erkennen, können Sie die Optionen **INTO** und **SET** weglassen.

RECEIVE (APPC)

Empfängt Daten bei einem formatfreien APPC-Datenaustausch.

RECEIVE (APPC)



Bedingungen: EOC, INVREQ, LENGERR, NOTALLOC, SIGNAL, TERMERR

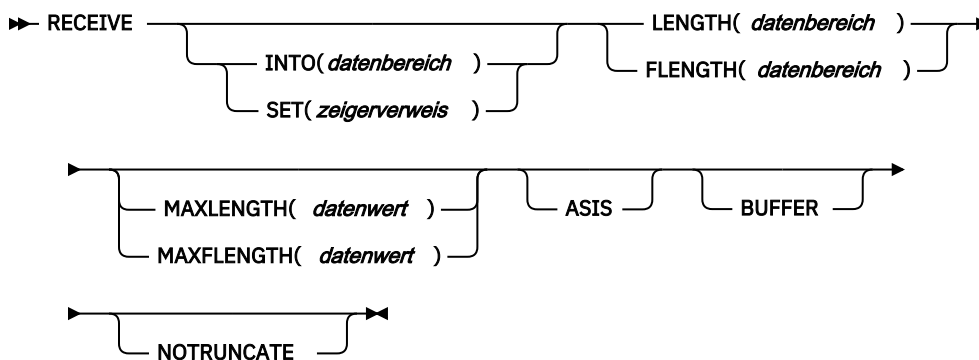
Beschreibung

Der Befehl **RECEIVE** empfängt bei einem formatfreien APPC-Datenaustausch Daten vom Dialogpartner.

RECEIVE (LUTYPE2/LUTYPE3)

Empfängt Daten von einer logischen Einheit mit 3270-Anzeige (LUTYPE2) oder mit 3270-Drucker (LUTYPE3).

RECEIVE (LUTYPE2/LUTYPE3)



Bedingungen: EOC, INVREQ, LENGERR, TERMERR

Beschreibung

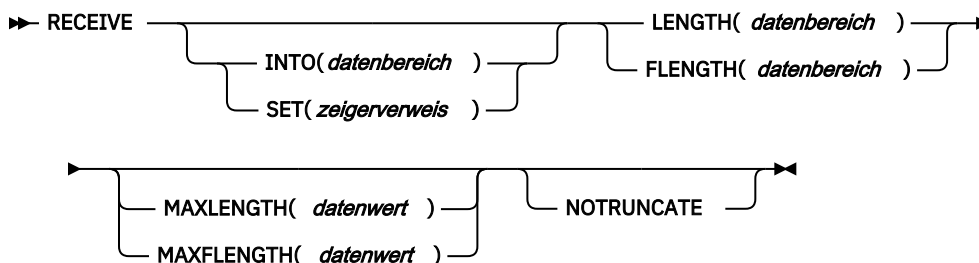
Der Befehl **RECEIVE** empfängt Daten vom Terminal.

Falls Daten empfangen werden sollen, müssen Sie entweder die Option **INTO** oder die Option **SET** angeben. Wenn ein Befehl **RECEIVE** ausschließlich zu dem Zweck ausgegeben wird, ein AID-Zeichen zu erkennen (und die Option **BUFFER** nicht angegeben wurde), können Sie die Optionen **INTO** und **SET** weglassen.

RECEIVE (LUTYPE4)

Empfängt Daten von einer logischen Einheit mit LU-Typ 4.

RECEIVE (LUTYPE4)



Bedingungen: EOC, EODS, INBFMH, INVREQ, LENGERR, SIGNAL, TERMERR

Beschreibung

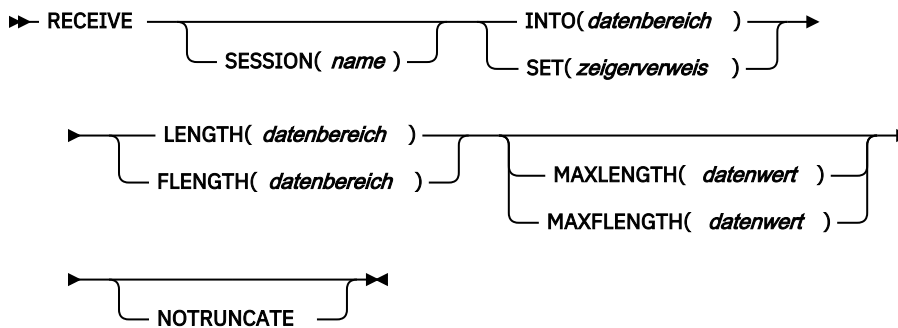
Der Befehl **RECEIVE** empfängt Daten vom Terminal.

Falls Daten empfangen werden sollen, müssen Sie entweder die Option **INTO** oder die Option **SET** angeben. Wenn ein Befehl **RECEIVE** ausschließlich zu dem Zweck ausgegeben wird, ein AID-Zeichen zu erkennen, können Sie die Optionen **INTO** und **SET** weglassen.

RECEIVE (LUTYPE6.1)

Empfängt Daten in einer Sitzung mit LU-Typ 6.1.

RECEIVE (LUTYPE6.1)



Bedingungen: EOC, INBFMH, INVREQ, LENGERR, NOTALLOC, SIGNAL, TERMERR

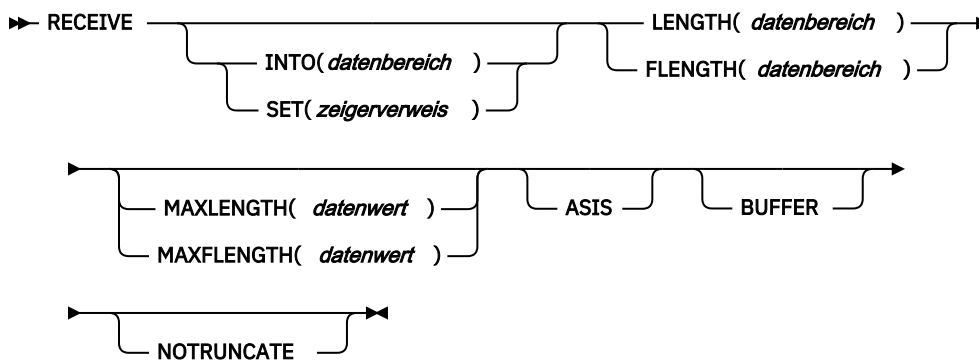
Beschreibung

Der Befehl **RECEIVE** empfängt bei einem Datenaustausch mit LU-Typ 6.1 Daten vom Dialogpartner.

RECEIVE (logische 3270-Einheit)

Empfängt Daten von einer logischen 3270-Einheit.

RECEIVE (logische 3270-Einheit)



Bedingungen: INVREQ, LENGERR, TERMERR

Beschreibung

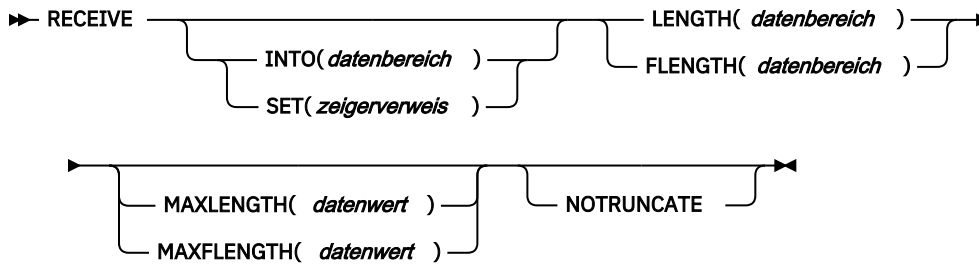
Der Befehl **RECEIVE** empfängt Daten von einem Terminal.

Falls Daten empfangen werden sollen, müssen Sie entweder die Option **INTO** oder die Option **SET** angeben. Wenn ein Befehl **RECEIVE** ausschließlich zu dem Zweck ausgegeben wird, ein AID-Zeichen zu erkennen (und die Option **BUFFER** nicht angegeben wurde), können Sie die Optionen **INTO** und **SET** weglassen.

RECEIVE (3600-Pipeline)

Empfängt Anfangseingabedaten von einer logischen Einheit mit 3600-Pipeline. Nachfolgende **RECEIVE**-Befehle für weitere Eingabedaten sind nicht zulässig.

RECEIVE (3600-Pipeline)



Bedingungen: INVREQ, LENGERR, NOTALLOC

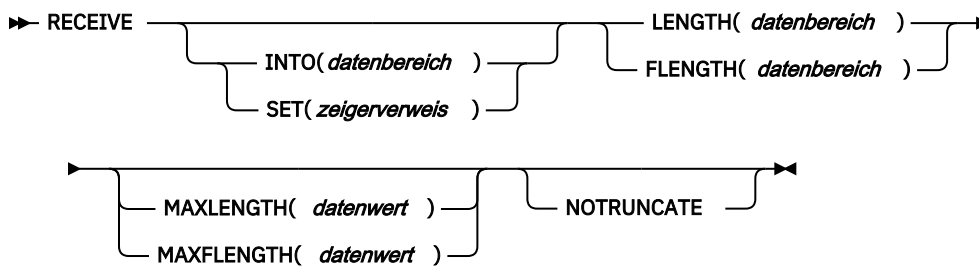
Beschreibung

Falls Daten empfangen werden sollen, müssen Sie entweder die Option INTO oder die Option SET angeben. Wenn ein Befehl RECEIVE ausschließlich zu dem Zweck ausgegeben wird, ein AID-Zeichen zu erkennen, können Sie die Optionen INTO und SET weglassen.

RECEIVE (3600-3601)

Empfängt Daten von einer logischen 3600- bzw. 3601-Einheit.

RECEIVE (3600-3601)



Bedingungen: EOC, EODS, INBFMH, INVREQ, LENGERR, SIGNAL, TERMERR

Beschreibung

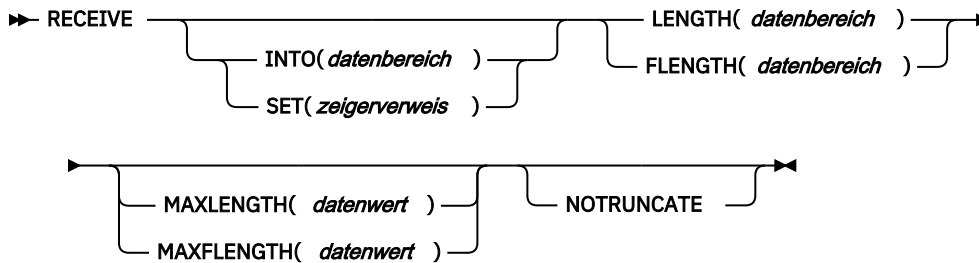
Der Befehl RECEIVE empfängt Daten vom Terminal. Dieses Format des Befehls RECEIVE gilt auch für das Werkskommunikationssystem IBM 3630 (Plant Communication System).

Falls Daten empfangen werden sollen, müssen Sie entweder die Option INTO oder die Option SET angeben. Wenn ein Befehl RECEIVE ausschließlich zu dem Zweck ausgegeben wird, ein AID-Zeichen zu erkennen, können Sie die Optionen INTO und SET weglassen.

RECEIVE (3600-3614)

Empfängt Daten von einer logischen 3600- bzw. 3614-Einheit.

RECEIVE (3600-3614)



Bedingungen: INVREQ, LENGERR, TERMERR

Beschreibung

Der Befehl RECEIVE empfängt Daten vom Terminal.

Das Datenstrom- und Kommunikationsformat, das zwischen einem CICS-Anwendungsprogramm und einer 3614-Einheit verwendet wird, wird durch die 3614-Einheit bestimmt. Das Anwendungsprogramm ist somit bei der Verarbeitung der 3614-Kommunikation einheitenabhängig.

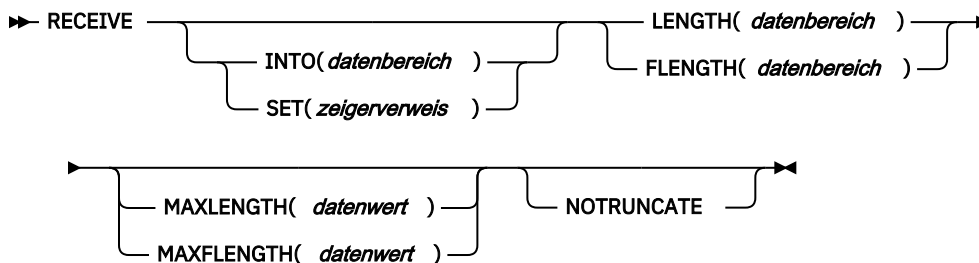
Weitere Informationen zum Entwerfen von 3614-Anwendungsprogrammen für CICS enthält das Handbuch zum System *IBM 4700/3600/3630*.

Falls Daten empfangen werden sollen, müssen Sie entweder die Option INTO oder die Option SET angeben. Wenn ein Befehl RECEIVE ausschließlich zu dem Zweck ausgegeben wird, ein AID-Zeichen zu erkennen, können Sie die Optionen INTO und SET weglassen.

RECEIVE (3650)

Empfängt Daten von logischen 3650-Einheiten.

RECEIVE (3650)



Bedingungen: EOC, EODS, INBFMH, INVREQ, LENGERR, TERMERR

Beschreibung

Der Befehl RECEIVE empfängt Daten vom Terminal. Dieses Format des Befehls RECEIVE gilt auch für die folgenden 3650-Einheiten:

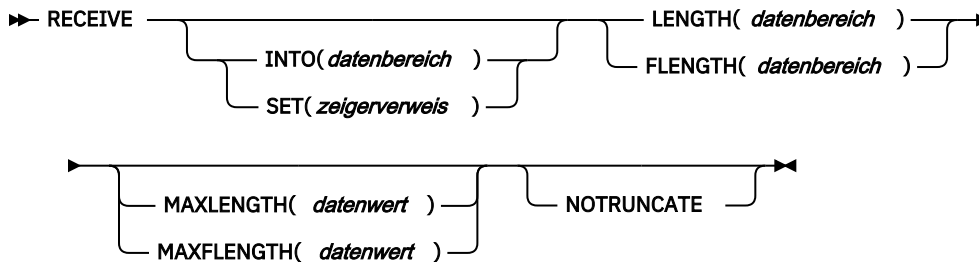
- Logische Einheit mit Interpreter
- Logische Einheit mit 3270-Host-Dialog
- Logische Einheit mit 3653-Host-Dialog
- Logische Einheit mit 3650- bzw. 3680-Befehlsprozessor

Falls Daten empfangen werden sollen, müssen Sie entweder die Option INTO oder die Option SET angeben. Wenn ein Befehl RECEIVE ausschließlich zu dem Zweck ausgegeben wird, ein AID-Zeichen zu erkennen, können Sie die Optionen INTO und SET weglassen.

RECEIVE (logische 3767-Einheit)

Empfängt Daten von einer interaktiven logischen 3767-Einheit.

RECEIVE (logische 3767-Einheit)



Bedingungen: EOC, INVREQ, LENGERR, SIGNAL, TERMERR

Beschreibung

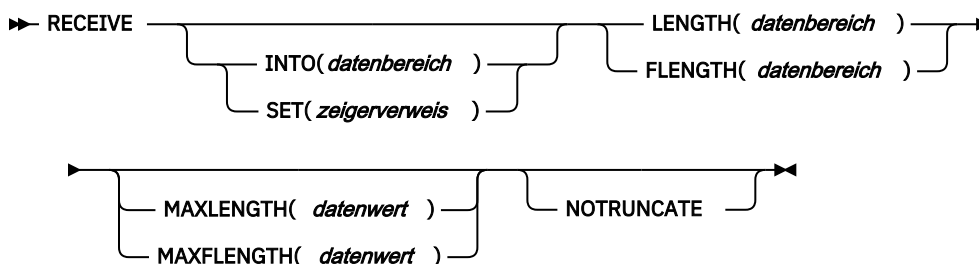
Der Befehl RECEIVE empfängt Daten vom Terminal. Dieses Format des Befehls RECEIVE gilt auch für die interaktive logische 3770-Einheit.

Falls Daten empfangen werden sollen, müssen Sie entweder die Option INTO oder die Option SET angeben. Wenn ein Befehl RECEIVE ausschließlich zu dem Zweck ausgegeben wird, ein AID-Zeichen zu erkennen, können Sie die Optionen INTO und SET weglassen.

RECEIVE (3770)

Empfängt Daten von einer logischen Einheit mit 3770-Stapel.

RECEIVE (3770)



Bedingungen: EOC, EODS, INBFMH, INVREQ, LENGERR, SIGNAL, TERMERR

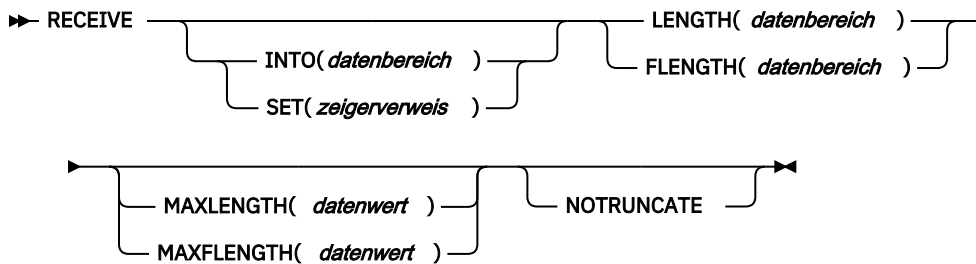
Beschreibung

Der Befehl RECEIVE empfängt Daten vom Terminal. Falls Daten empfangen werden sollen, müssen Sie entweder die Option INTO oder die Option SET angeben. Wenn ein Befehl RECEIVE ausschließlich zu dem Zweck ausgegeben wird, ein AID-Zeichen zu erkennen, können Sie die Optionen INTO und SET weglassen.

RECEIVE (3790 mit vollem Funktionsumfang oder Anfrage)

Empfängt Daten von einer logischen 3790-Einheit mit vollem Funktionsumfang oder Anfrage.

RECEIVE (3790 mit vollem Funktionsumfang oder Anfrage)



Bedingungen: EOC, EODS, INBFMH, INVREQ, LENGERR, SIGNAL, TERMERR

Beschreibung

Der Befehl RECEIVE empfängt Daten vom Terminal. Dieses Format des Befehls RECEIVE gilt auch für die folgenden Einheiten:

- Logische 3650- bzw. 3680-Einheit mit vollem Funktionsumfang
- Logische 3770-Einheit mit vollem Funktionsumfang

Falls Daten empfangen werden sollen, müssen Sie entweder die Option INTO oder die Option SET angeben. Wenn ein Befehl RECEIVE ausschließlich zu dem Zweck ausgegeben wird, ein AID-Zeichen zu erkennen, können Sie die Optionen INTO und SET weglassen.

RECEIVE: Optionen für z/OS Communications Server

In diesem Abschnitt sind allgemeine Optionen beschrieben, die für den Befehl RECEIVE (bei z/OS Communications Server) verwendet werden.

Optionen

ASIS

Gibt an, dass Kleinbuchstaben im 3270-Eingabedatenstrom nicht in Großbuchstaben umgesetzt werden; hierdurch ist die aktuelle Task in der Lage, eine Nachricht zu empfangen, die Daten sowohl in Klein- als auch in Großschreibung enthält.

Auf den ersten Befehl RECEIVE einer Transaktion hat diese Option keinen Einfluss, weil die Terminalsteuerung eine Operation READ INITIAL ausführt und die Standardeinstellungen des Terminals für die Umsetzung der Operationsdaten verwendet.

Falls die Anzeige Daten enthält, bevor eine Transaktion gestartet wird, hat diese Option keine Auswirkungen. Diese Daten werden gelesen und zur Vorbereitung der nächsten Task umgesetzt. Der erste Befehl RECEIVE in dieser Task ruft dann die umgesetzten Daten ab.

Anmerkung: Falls Sie ein Katakana-Terminal verwenden, werden möglicherweise einige Nachrichten angezeigt, in denen englische und Katakana-Zeichen kombiniert sind. Dies liegt daran, dass Katakana-Terminals keine Ausgabe in Groß-/Kleinschreibung anzeigen können. Großbuchstaben im Datenstrom werden als englische Großbuchstaben angezeigt, Kleinbuchstaben jedoch als Katakana-Zeichen. Bitten Sie in einem solchen Fall Ihren Systemprogrammierer, in den Systeminitialisierungsparametern MSGCASE=UPPER anzugeben, damit diese Nachrichten ausschließlich Großbuchstaben enthalten.

BUFFER

Gibt an, dass der Inhalt des 3270-Puffers ab Pufferposition 1 gelesen und der Lesevorgang fortgesetzt werden soll, bis der gesamte Inhalt des Puffers gelesen wurde. Alle Zeichen- und Attributsequenzen (inklusive Nullen) treten im Eingabedatenstrom in derselben Reihenfolge wie im 3270-Puffer auf.

CONVID(name)

Gibt den Dialog an, auf den sich der Befehl bezieht. Der vier Zeichen lange Name gibt entweder das durch einen zuvor ausgeführten Befehl ALLOCATE im Feld EIBRSRCE des EIB zurückgegebene Token oder aber das Token an, das die Hauptsitzung darstellt (und durch einen zuvor ausgeführten Befehl ASSIGN zurückgegeben wurde).

Zur Kompatibilität mit früheren Releases wird die Option SESSION als Synonym für CONVID akzeptiert. Neue Programme sollten jedoch CONVID verwenden.

Falls diese Option nicht angegeben wird, wird die Hauptfunktion verwendet.

FLENGTH(datenbereich)

Dies ist eine Alternative zu LENGTH. Aus architekturabhängigen Gründen ist diese Option bei allen terminalbezogenen Befehlen RECEIVE auf 32 K begrenzt.

INTO(datenbereich)

Gibt das empfangende Feld für die Daten an, die aus der logischen Einheit oder dem Terminal gelesen wurden, bzw. den Zieldatenbereich der Anwendung, in dem die Daten aus dem Anwendungsprogramm am anderen Ende der Verbindung für den aktuellen Datenaustausch empfangen werden sollen.

LENGTH(datenbereich)

Gibt als Halbwort-Binärwert die Länge der empfangenen Daten an.

Falls Sie die Option INTO, jedoch nicht die Option MAXLENGTH angeben, muss das Argument ein Datenbereich sein, der die vom Programm akzeptierte maximale Länge angibt. Falls der angegebene Wert kleiner als Null ist, wird Null angenommen.

Bei Angabe der Option SET muss das Argument ein Datenbereich sein. Nachdem die Daten empfangen wurden, wird der Datenbereich auf die Länge der Daten gesetzt.

MAXFLENGTH(datenwert)

Dies ist eine Vollwortalternative zu MAXLENGTH.

MAXLENGTH(datenwert)

Gibt (als Halbwort-Binärwert) die maximale Menge der Daten an, die CICS wiederherstellen soll. Bei Angabe von INTO setzt MAXLENGTH die Verwendung von LENGTH als Eingabe für CICS außer Kraft. Bei Angabe von SET bietet MAXLENGTH eine Möglichkeit für das Programm, das jeweils empfangene Datenvolumen zu begrenzen.

Falls die Länge der Daten den angegebenen Wert überschreitet und die Option NOTRUNCATE nicht angegeben ist, werden die Daten auf diesen Wert abgeschnitten und die Bedingung LENGERR tritt auf. Der in der Option LENGTH angegebene Datenbereich wird auf die ursprüngliche Länge der Daten gesetzt.

Falls die Länge der Daten den angegebenen Wert überschreitet und die Option NOTRUNCATE angegeben ist, bewahrt CICS die verbleibenden Daten auf und verwendet sie zur Erfüllung nachfolgender Befehle RECEIVE. Der in der Option LENGTH angegebene Datenbereich wird auf die Länge der zurückgegebenen Daten gesetzt.

Falls diese Option weggelassen wird, wird der in der Option LENGTH angegebene Wert angenommen.

NOTRUNCATE

Gibt an, dass bei einer Überschreitung der angeforderten Länge durch die verfügbaren Daten die übrigen Daten nicht gelöscht, sondern zum Abruf durch nachfolgende Befehle RECEIVE aufbewahrt werden sollen.

SESSION(name)

Gibt die symbolische ID (1 bis 4 Zeichen) eines Eintrags in der Terminalsteuertabelle für eine Sitzung an. Diese Option gibt die zu verwendende alternative Funktion an. Falls sie nicht angegeben ist, wird die Hauptfunktion für die Task verwendet.

SET(zeigerverweis)

Gibt den Zeigerverweis an, der auf die Adresse der Daten gesetzt werden soll, die aus der logischen Einheit oder dem Terminal bzw. der Partnertransaktion gelesen wurden. Der Zeigerverweis ist bis zum nächsten Befehl RECEIVE oder bis zum Ende der Task gültig.

Wenn dem Anwendungsprogramm DATALOCATION(ANY) zugeordnet ist, kann die Adresse der Daten ober- oder unterhalb der 16-MB-Grenze liegen.

Wenn dem Anwendungsprogramm DATALOCATION(BELOW) zugeordnet ist und sich die Daten oberhalb der 16-MB-Grenze befinden, werden die Daten an eine Adresse unterhalb der 16-MB-Grenze kopiert und die Adresse dieser Kopie wird zurückgegeben.

Wenn für die aktive Task TASKDATAKEY(USER) angegeben wird und der Speicherschutz aktiv ist, befinden sich die zurückgegebenen Daten in einem Benutzerschlüssel. Wenn für die aktive Task TASKDATAKEY(CICS) angegeben wird und der Speicherschutz aktiv ist, befinden sich die zurückgegebenen Daten in einem CICS-Schlüssel.

STATE(cvda)

Ruft den Status des aktuellen Datenaustausches ab. Die folgenden CVDA-Werte können von CICS zurückgegeben werden:

- ALLOCATED
- CONFFREE
- CONFRECEIVE
- CONFSEND
- FREE
- PENDFREE
- PENDRECEIVE
- RECEIVE
- ROLLBACK
- SEND
- SYNCFREE
- SYNCRECEIVE
- SYNCSEND

Bedingungen

Einige der folgenden Bedingungen können in Kombination mit anderen Bedingungen auftreten. CICS überprüft in der folgenden Reihenfolge, ob diese Bedingungen vorliegen:

1. EODS
2. INBFMH
3. EOC

Falls mehrere Bedingungen auftreten, wird nur die erste Bedingung an das Anwendungsprogramm übergeben. EIBRCODE wird jedoch so festgelegt, dass alle aufgetretenen Bedingungen angegeben sind.

06 EOC

Tritt auf, wenn die Anforderung/Antwort-Einheit mit festgelegtem Anzeiger für das Kettenende empfangen wird. Das Feld EIBEOC gibt diese Bedingung ebenfalls an.

Standardaktion: Die Bedingung wird ignoriert.

05 EODS (nur logische Einheit mit Interpreter)

Tritt auf, wenn ein Dateiendeanzeiger empfangen wird.

Standardaktion: Die Task wird abnormal beendet.

07 INBFMH

Tritt auf, falls eine Anforderung/Antwort-Einheit einen Funktionsverwaltungsheader (FMH) enthält. Das Feld EIBFMH enthält diesen Anzeiger und sollte vorzugsweise anstelle des Feldes INBFMH verwendet werden. Mit dem Befehl IGNORE CONDITION kann die Bedingung ignoriert werden.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

RESP2-Werte:

200

Eine Serveranwendung für Distributed Program Link hat in der Option CONVID die Sitzung für die Funktionsverlagerung (ihre Hauptfunktion) angegeben.

Tritt außerdem in jedem der folgenden Fälle auf (RESP2 nicht festgelegt):

- Der Befehl wird für einen APPC-Datenaustausch verwendet, der nicht die EXEC-Schnittstelle von CICS nutzt oder kein formatfreier Datenaustausch ist.

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

Tritt auf, falls Daten durch CICS gelöscht werden, weil ihre Länge den vom Programm akzeptierten Höchstwert überschreiten und die Option NOTRUNCATE nicht angegeben ist.

Standardaktion: Die Task wird abnormal beendet.

61 NOTALLOC

Tritt auf, falls der Befehl RECEIVE durch eine Transaktion ausgegeben wird, die durch den Befehl START als Nichtterminaltask gestartet wurde, oder falls der Wert für CONVID oder die im Befehl angegebene Funktion nicht zu einem Datenaustausch gehört, dessen Eigner die Anwendung ist.

Standardaktion: Die Task wird abnormal beendet.

24 SIGNAL

Tritt auf, wenn ein eingehender Befehl SIGNAL für die Datenflusssteuerung von einer Partnertransaktion empfangen wird. Beim Empfang eines eingehenden Signals wird das Feld EIBSIG immer festgelegt.

Standardaktion: Die Bedingung wird ignoriert.

81 TERMERR

Tritt für einen sitzungs- oder terminalbezogenen Fehler auf. Außer FREE bewirken alle anderen Aktionen für diesen Datenaustausch einen Abbruch des Typs ATCV.

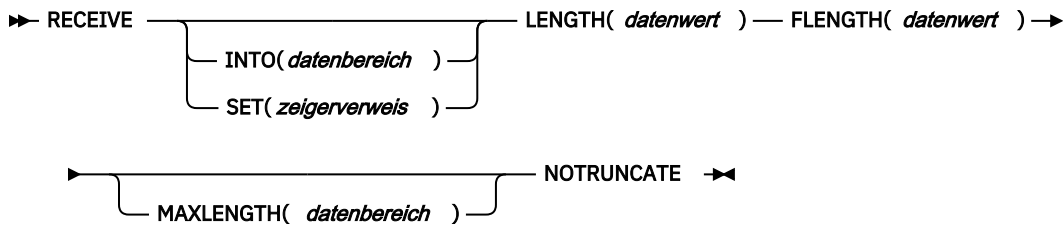
Diese Bedingung kann durch eine Anforderung CANCEL TASK durch ein Knotenfehlerprogramm des Benutzers verursacht werden, falls für die Task eine Terminalsteuerungsanforderung aussteht, wenn der Sitzungsfehler durch das Programm des Knotens für abnormale Bedingungen abgewickelt wird.

Standardaktion: Die Task wird abnormal mit dem Abbruchcode ATNI beendet.

RECEIVE (Standardformat ohne z/OS Communications Server)

Empfängt Daten von einer Task, die nicht an ein Terminal angehängt ist.

RECEIVE (Standardformat)



Bedingungen: INVREQ, LENGERR, NOTALLOC

Beschreibung

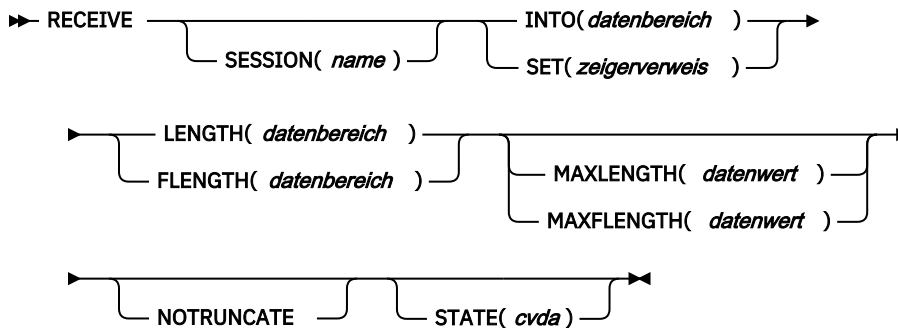
Diese Form des Befehls **RECEIVE** wird von allen von CICS unterstützten Terminals verwendet, für die die anderen **RECEIVE**-Beschreibungen nicht geeignet sind.

Falls Daten empfangen werden sollen, müssen Sie entweder die Option **INTO** oder die Option **SET** angeben. Wenn ein Befehl **RECEIVE** ausschließlich zu dem Zweck ausgegeben wird, ein AID-Zeichen zu erkennen, können Sie die Optionen **INTO** und **SET** weglassen.

RECEIVE (MRO)

Empfängt Daten in einem MRO-Dialog.

RECEIVE (MRO)



Bedingungen: EOC, INBFMH, INVREQ, LENGERR, NOTALLOC, TERMERR

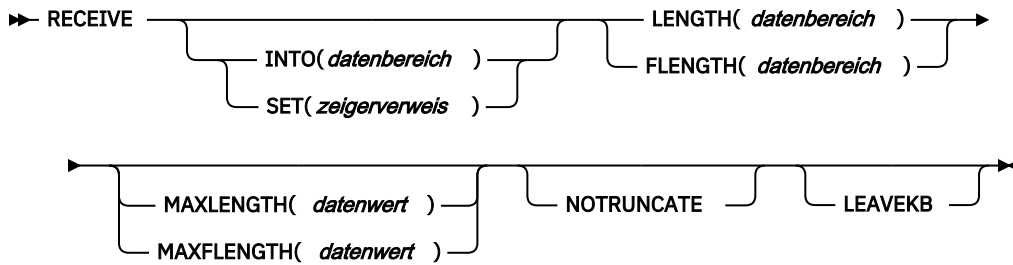
Beschreibung

Der Befehl **RECEIVE** empfängt bei einem Datenaustausch im Mehrregionenbetrieb (Multi Region Operation, MRO) Daten vom Dialogpartner.

RECEIVE (2260)

Empfängt Daten von einer 2260- oder 2265-Anzeigestation.

RECEIVE (2260)



Bedingungen: INVREQ, LENGERR

Beschreibung

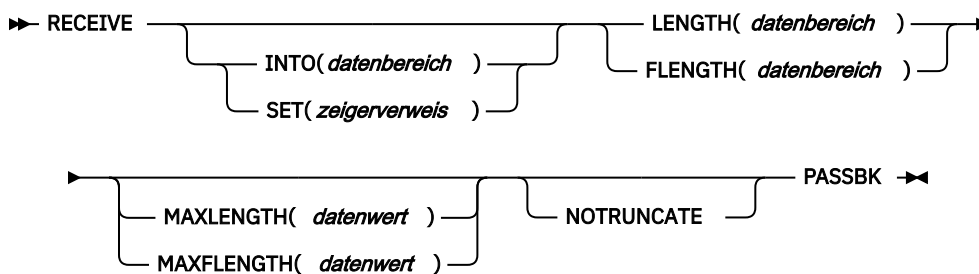
Der Befehl RECEIVE empfängt Daten vom Terminal.

Falls Daten empfangen werden sollen, müssen Sie entweder die Option INTO oder die Option SET angeben. Wenn ein Befehl RECEIVE ausschließlich zu dem Zweck ausgegeben wird, ein AID-Zeichen zu erkennen, können Sie die Optionen INTO und SET weglassen.

RECEIVE (2980)

Empfängt Daten von einem Terminalsystem IBM 2980 für allgemeine Bankgeschäfte.

RECEIVE (2980)



Bedingungen: INVREQ, LENGERR, NOPASSBKRD

Beschreibung

Der Befehl RECEIVE empfängt Daten vom Terminal.

Falls Daten empfangen werden sollen, müssen Sie entweder die Option INTO oder die Option SET angeben. Wenn ein Befehl RECEIVE ausschließlich zu dem Zweck ausgegeben wird, ein AID-Zeichen zu erkennen, können Sie die Optionen INTO und SET weglassen.

Sparbuchsteuerung

Alle Eingabe- und Ausgabeanforderungen für den Sparbuchbereich eines Systems IBM 2980 sind vom Vorhandensein eines Sparbuchs abhängig. Mit der Option PASSBK wird angegeben, dass die Kommunikation mit einem Sparbuch erfolgt. Die Bedingungen NOPASSBKRD (RECEIVE) und NOPASSBKWR (SEND) treten bei Eingabe- bzw. Ausgabeanforderungen auf, wenn ein Sparbuch nicht vorhanden ist. Diese Bedingungen können durch einen Befehl HANDLE CONDITION und entsprechende Behandlungsroutinen abgewickelt werden.

Falls das Sparbuch in einer Eingabeanforderung vorhanden ist, schreibt das Anwendungsprogramm Daten generell in den Sparbuchbereich zurück, um das Sparbuch zu aktualisieren. Falls die Bedingung NOPASSBKWR auftritt, lässt CICS die sofortige Ausgabe am Terminal zu. In einer Routine für die Bedingung NOPASSBKWR sollte das Anwendungsprogramm eine Fehlermeldung an den Journalbereich des Terminals senden, um den Bediener des Systems IBM 2980 über diese Fehlerbedingung zu informieren. Damit der Bediener das erforderliche Sparbuch einfügen kann, veranlasst CICS einen Wartestatus von 23,5 Sekunden, bevor die Transaktion fortgesetzt wird.

Sobald das Anwendungsprogramm nach dem Senden der Fehlermeldung von CICS die Steuerung zurückerhält, kann es erneut versuchen, das Sparbuch zu aktualisieren, wenn es sichergestellt hat, dass das Druckelement im Sparbuchbereich richtig positioniert ist. Hierzu werden im Allgemeinen zwei Zeilenschaltungen ausgegeben, auf die so viele Tabstopps folgen, wie erforderlich sind, um das Druckelement an die korrekte Position zu verschieben.

Falls bei dem Versuch, Daten in den Sparbuchbereich zu schreiben, die Bedingung NOPASSBKWR auftritt, kann das Anwendungsprogramm eine weitere Fehlermeldung senden oder eine alternative Aktion ausführen (beispielsweise das Terminal außer Betrieb nehmen). Das Vorhandensein des Prüffunktionsschlüssels auf einer 2980-Verwaltungsstation Modell 2 wird durch den Befehl SEND PASSBK gesteuert und kann auf ähnliche Weise wie zuvor beschrieben verwendet werden.

Ausgabesteuerung

Die Übertragungseinheit für ein System IBM 2980 wird als **Segment** bezeichnet. Für den Sparbuch- und den Journalbereich lässt CICS jedoch zu, dass ein Anwendungsprogramm Nachrichten sendet, die die Puffergröße überschreiten. Für den Sparbuchbereich ist die maximale Länge einer Nachricht auf eine Zeile eines Sparbuchs begrenzt, um Leerräume (*Indexierung*) nach dem Ende des Sparbuchs zu vermeiden. Für den Journalbereich ist die maximale Nachrichtenlänge in der Option LENGTH des Befehls SEND angegeben.

Beispiel: Ein 2972-Puffer hat die Größe von 48 Zeichen und der Druckbereich für ein Sparbuch umfasst auf einer 2980-Bankschalterstation Modell 4 100 Zeichen pro Zeile. Das Anwendungsprogramm kann an diesen Bereich eine Nachricht von 100 Zeichen senden; CICS segmentiert die Nachricht, um sie an die Puffergröße anzupassen. Das Anwendungsprogramm muss das Zeichen für die Sparbuchindexierung (X'25') als **letztes** Zeichen in einer Eingabeanforderung an den Sparbuchbereich senden. Hierdurch wird die Indexierung des Sparbuchs gesteuert und auf diese Weise die positive Steuerung für das Vorhandensein des Sparbuchs erreicht.

Wenn eine Nachricht eingebettete Zeichen für die Sparbuchindexierung enthält und aufgrund der Nachrichtenlänge eine Segmentierung erforderlich ist, wird die Ausgabe beendet, falls das Sparbuch Leerräume enthält, die über das Ende des Sparbuchs hinausgehen; die verbleibenden Segmente werden nicht gedruckt.

Ausgabe in allgemeinen Puffer

Mit dem Befehl SEND CBUFF werden Daten an einen allgemeinen Puffer übertragen. Die Daten werden in den Zeichensatz des empfangenden 2980-Modells umgesetzt. Falls mit dem 2972-Controller mehrere 2980-Modelltypen verbunden sind, werden die Längen abgeschnitten, falls sie die Puffergröße überschreiten.

DFH2980-Struktur

Die DFH2980-Struktur enthält Konstanten, die nur beim Schreiben von COBOL- oder PL/I-Anwendungsprogrammen für das System IBM 2980 verwendet werden können. Die Struktur wird durch das Kopieren von DFH2980 in das Anwendungsprogramm erhalten.

Bei COBOL wird DFH2980 in den Arbeitsspeicherabschnitt kopiert; bei PL/I wird DFH2980 unter Verwendung einer Anweisung %INCLUDE einbezogen.

Die Stationskennung wird im Feld STATIONID angegeben, dessen Wert mit dem Befehl ASSIGN ermittelt werden muss. Um zu testen, ob eine normale oder eine alternative Station verwendet wird, wird das Feld STATIONID mit Werten verglichen, die in DFH2980 vordefiniert sind. Die Werte lauten wie folgt:

STATION-**n**-A oder STATION-**n**-N-

STATION_**n**_A oder STATION_**n**_N

Hierbei ist **n** eine ganze Zahl (0 bis 9); A und N stehen für 'alternative Station' und 'normale Station'. (Das Umbruchssymbol ist bei COBOL der Bindestrich (-) und bei PL/I das Unterstreichungszeichen (_).)

Die Bankautomatkennung für eine 2980-Bankschalterstation Modell 4 ist im Zeichenfeld TELLERID angegeben, das eine Länge von einem Byte hat. Der Wert für TELLERID muss mit einem Befehl ASSIGN ermittelt werden.

In das Anwendungsprogramm müssen Tabulatorzeichen (X'05') einbezogen werden. Die Anzahl der Tabulatorstopps, die zum Positionieren des Druckelements an der ersten Position eines Sparsbuchbereichs erforderlich ist, wird im Feld NUMTAB angegeben. Der Wert von NUMTAB muss mit einem Befehl ASSIGN ermittelt werden. Der Wert von NUMTAB wird durch den Systemprogrammierer angegeben und kann bei jedem Terminal unterschiedlich sein.

Andere Tabulatorzeichen werden zur Steuerung der Formatierung wie benötigt eingefügt.

Jeder der DFH2980-Werte TAB-ZERO bis TAB-NINE für COBOL und PL/I kann mit dem Wert von NUMTAB verglichen werden, um die Anzahl der Tabulatorzeichen zu ermitteln, die in einer Ausgabenachricht eingefügt werden müssen, um die korrekte Positionierung des Druckelements zu erreichen. Das Tabulatorzeichen ist in DFH2980 als TABCHAR enthalten.

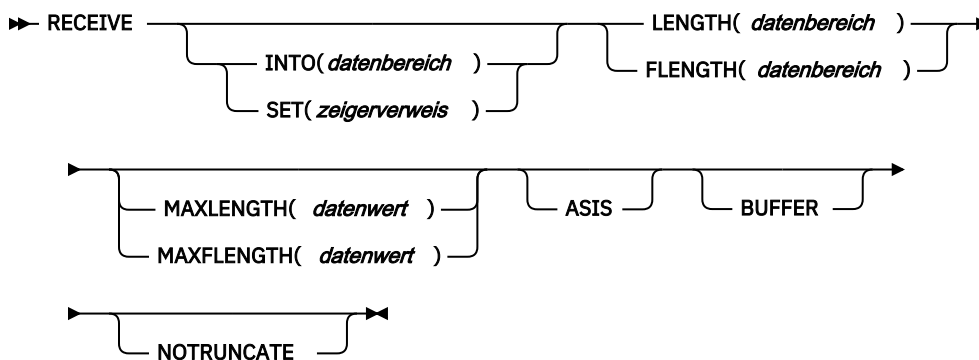
In DFH2980 sind insgesamt 30 Sonderzeichen definiert. 23 dieser Zeichen können durch den Namen SPECCHAR-# oder SPECCHAR_# (bei American National Standard COBOL bzw. PL/I) referenziert werden, wobei # für eine ganze Zahl (0 bis 22) steht. Die sieben weiteren Zeichen sind mit Namen definiert, die ihre Verwendung abbilden (z. B. TABCHAR).

Einige andere in DFH2980 definierte Zeichen wie HOLDPCF oder TCTTEPCR sind für den Einsatz in Anwendungsprogrammen gedacht, die CICS-Makros verwenden, und sollten in Anwendungsprogrammen, die CICS-Befehle verwenden, nicht erforderlich sein.

RECEIVE (3790- bzw. 3270-Anzeige)

Empfängt Daten von einer logischen Einheit mit 3790- bzw. 3270-Anzeige.

RECEIVE (3790- bzw. 3270-Anzeige)



Bedingungen: INVREQ, LENGERR, TERMERR

Beschreibung

Der Befehl RECEIVE empfängt Daten vom Terminal.

Falls Daten empfangen werden sollen, müssen Sie entweder die Option INTO oder die Option SET angeben. Wenn ein Befehl RECEIVE ausschließlich zu dem Zweck ausgegeben wird, ein AID-Zeichen zu erkennen (und die Option BUFFER nicht angegeben wurde), können Sie die Optionen INTO und SET weglassen.

RECEIVE: Optionen ohne z/OS Communications Server

In diesem Abschnitt sind allgemeine Optionen beschrieben, die für den Befehl RECEIVE verwendet werden (nicht bei z/OS Communications Server).

Optionen

ASIS

Gibt an, dass die Ausgabe im Transparenzmodus gesendet werden soll, also ohne Erkennung von Steuerzeichen und unter Akzeptanz aller 256 möglichen 8-Bit-Kombinationen als gültige und übertragbare Daten.

Auf den ersten Befehl RECEIVE einer Transaktion hat diese Option keinen Einfluss, weil die Terminalsteuerung eine Operation READ INITIAL ausführt und die Standardeinstellungen des Terminals für die Datenumsetzung verwendet.

Falls die Anzeige Daten enthält, bevor eine Transaktion gestartet wird, hat diese Option keine Auswirkungen. Diese Daten werden gelesen und zur Vorbereitung der nächsten Task umgesetzt. Der erste Befehl RECEIVE in dieser Task ruft dann die umgesetzten Daten ab.

Anmerkung: Falls Sie ein Katakana-Terminal verwenden, werden möglicherweise einige Nachrichten angezeigt, in denen englische und Katakana-Zeichen kombiniert sind. Dies liegt daran, dass Katakana-Terminals keine Ausgabe in Groß-/Kleinschreibung anzeigen können. Großbuchstaben im Datenstrom werden als englische Großbuchstaben angezeigt, Kleinbuchstaben jedoch als Katakana-Zeichen. Bitten Sie in einem solchen Fall Ihren Systemprogrammierer, in den Systeminitialisierungsparametern MSGCASE=UPPER anzugeben, damit diese Nachrichten ausschließlich Großbuchstaben enthalten.

BUFFER

Gibt an, dass der Inhalt des 3270-Puffers ab Pufferposition 1 gelesen und der Lesevorgang fortgesetzt werden soll, bis der gesamte Inhalt des Puffers gelesen wurde. Alle Zeichen- und Attributsequenzen (inklusive Nullen) treten im Eingabedatenstrom in derselben Reihenfolge wie im 3270-Puffer auf.

LENGTH(datenbereich)

Dies ist eine Vollwortalternative zu LENGTH.

INTO(datenbereich)

Gibt das empfangende Feld für die Daten an, die aus dem Terminal oder der logischen Einheit gelesen wurden, oder den Zielbereich des Anwendungsprogramms, das die Daten aus dem Anwendungsprogramm am anderen Ende der Verbindung für den aktuellen Datenaustausch empfängt.

Falls Sie die Option INTO, jedoch nicht die Option MAXLENGTH angeben, muss das Argument für die Option LENGTH ein Datenbereich sein, der die vom Programm akzeptierte maximale Länge angibt. Falls der angegebene Wert kleiner als Null ist, wird Null angenommen.

Falls die Länge der Daten den angegebenen Wert überschreitet, die Option NOTRUNCATE jedoch nicht angegeben ist, werden die Daten auf diesen Wert abgeschnitten und die Bedingung LENGERR tritt auf. Sobald die Daten empfangen wurden, wird der Datenbereich für die Option LENGTH auf die ursprüngliche Länge der Daten gesetzt.

LEAVEKB

Gibt an, dass die Tastatur nach Abschluss der Datenübertragung gesperrt bleibt.

LENGTH(datenbereich)

Gibt als Halbwort-Binärwert die Länge der übertragenen Daten an.

Falls Sie die Option INTO, jedoch nicht die Option MAXLENGTH angeben, muss das Argument ein Datenbereich sein, der die vom Programm akzeptierte maximale Länge angibt. Falls der angegebene Wert kleiner als Null ist, wird Null angenommen.

Falls die Länge der Daten den angegebenen Wert überschreitet, die Option NOTRUNCATE jedoch nicht angegeben ist, werden die Daten auf diesen Wert abgeschnitten und die Bedingung LENGERR tritt auf. Sobald die Daten empfangen wurden, wird der Datenbereich auf die ursprüngliche Länge der Daten gesetzt.

Bei Angabe der Option SET muss das Argument ein Datenbereich sein. Nachdem die Daten empfangen wurden, wird der Datenbereich auf die Länge der Daten gesetzt.

Eine Beschreibung einer sicheren Obergrenze finden Sie unter „[LENGTH-Optionen in CICS-Befehlen](#)“ auf Seite 10.

MAXLENGTH(datenwert)

Dies ist eine Vollwortalternative zu MAXLENGTH.

MAXLENGTH(datenwert)

Gibt (als Halbwort-Binärwert) die maximale Menge der Daten an, die CICS wiederherstellen soll. Bei Angabe von INTO setzt MAXLENGTH die Verwendung von LENGTH als Eingabe für CICS außer Kraft. Bei Angabe von SET bietet MAXLENGTH eine Möglichkeit für das Programm, das jeweils empfangene Datenvolumen zu begrenzen.

Falls der angegebene Wert kleiner als Null ist, wird Null angenommen.

Falls die Länge der Daten den angegebenen Wert überschreitet und die Option NOTRUNCATE nicht angegeben ist, werden die Daten auf diesen Wert abgeschnitten und die Bedingung LENGERR tritt auf. Der in der Option LENGTH angegebene Datenbereich wird auf die ursprüngliche Länge der Daten gesetzt.

Falls die Länge der Daten den angegebenen Wert überschreitet und die Option NOTRUNCATE angegeben ist, bewahrt CICS die verbleibenden Daten auf und verwendet sie zur Erfüllung nachfolgender Befehle RECEIVE. Der in der Option LENGTH angegebene Datenbereich wird auf die Länge der zurückgegebenen Daten gesetzt.

Falls diese Option weggelassen wird, wird der in der Option LENGTH angegebene Wert angenommen.

NOTRUNCATE

Gibt an, dass bei einer Überschreitung der angeforderten Länge durch die verfügbaren Daten die übrigen Daten nicht gelöscht, sondern zum Abruf durch nachfolgende Befehle RECEIVE aufbewahrt werden sollen.

PASSBK

Gibt an, dass die Kommunikation mit einem Sparbuch erfolgt.

PSEUDOBIN

Gibt an, dass die gelesenen Daten aus der pseudobinären Darstellung von System/7 in die hexadezimale Darstellung umgesetzt werden sollen.

SESSION(name)

Gibt die symbolische ID (1 bis 4 Zeichen) eines Eintrags in der Terminalsteuertabelle für eine Sitzung an. Diese Option gibt die zu verwendende alternative Funktion an. Falls sie nicht angegeben ist, wird die Hauptfunktion für die Task verwendet.

SET(zeigerverweis)

Gibt den Zeigerverweis an, der auf die Adresse der Daten gesetzt werden soll, die in einem MRO-Dialog vom Dialogpartner empfangen wurden. Der Zeigerverweis ist bis zum nächsten Befehl RECEIVE oder bis zum Ende der Task gültig.

Falls dem Anwendungsprogramm DATALOCATION(ANY) zugeordnet ist, kann die Adresse der Daten ober- oder unterhalb der 16-MB-Grenze liegen.

Falls dem Anwendungsprogramm DATALOCATION(BELOW) zugeordnet ist und sich die Daten oberhalb der 16-MB-Grenze befinden, werden die Daten an eine Adresse unterhalb der 16-MB-Grenze kopiert und die Adresse dieser Kopie wird zurückgegeben.

Falls für die aktive Task TASKDATAKEY(USER) angegeben und der Speicherschutz aktiv ist, befinden sich die zurückgegebenen Daten in einem Benutzerschlüssel. Wenn für die aktive Task TASKDATAKEY(CICS) angegeben wird und der Speicherschutz aktiv ist, befinden sich die zurückgegebenen Daten in einem CICS-Schlüssel.

Bei Angabe der Option SET muss das Argument für die Option LENGTH ein Datenbereich sein. Nachdem die Daten empfangen wurden, wird der Datenbereich auf die Länge der Daten gesetzt.

STATE(*cvda*)

Ruft den Status des aktuellen Datenaustausches ab. Die folgenden CVDA-Werte können von CICS zurückgegeben werden:

- ALLOCATED
- FREE
- PENDFREE
- RECEIVE
- ROLLBACK
- SEND
- SYNCFREE
- SYNCRECEIVE
- SYNCSEND

Bedingungen

Die folgenden Bedingungen können in Kombination mit anderen Bedingungen auftreten. CICS überprüft in der folgenden Reihenfolge, ob diese Bedingungen vorliegen:

1. INBFMH
2. EOC

Falls mehrere Bedingungen auftreten, wird nur die erste Bedingung an das Anwendungsprogramm übergeben. EIBRCODE wird jedoch so festgelegt, dass alle aufgetretenen Bedingungen angegeben sind.

08 ENDINPT

Tritt auf, wenn ein Anzeiger für das Ende der Eingabe (End-of-input, EOI) empfangen wird.

Standardaktion: Die Task wird abnormal beendet.

06 EOC

Tritt auf, wenn die Anforderung/Antwort-Einheit mit festgelegtem Anzeiger für das Kettenende empfangen wird. Das Feld EIBEOC enthält diesen Anzeiger ebenfalls.

Standardaktion: Die Bedingung wird ignoriert.

04 EOF

Tritt auf, wenn ein Dateiendeanzeiger (End-of-file, EOF) empfangen wird.

Standardaktion: Die Task wird abnormal beendet.

07 INBFMH

Tritt auf, falls eine Anforderung/Antwort-Einheit einen Funktionsverwaltungsheader (FMH) enthält. Das Feld EIBFMH enthält diesen Anzeiger und sollte vorzugsweise anstelle des Feldes INBFMH verwendet werden. Der Befehl IGNORE CONDITION kann verwendet werden, um die Bedingung zu ignorieren.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

RESP2-Werte:

200

Eine Serveranwendung für Distributed Program Link hat in der Option CONVID die Sitzung für die Funktionsverlagerung (ihre Hauptfunktion) angegeben.

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

Tritt auf, falls Daten durch CICS gelöscht werden, weil ihre Länge den vom Programm akzeptierten Höchstwert überschreiten und die Option NOTRUNCATE nicht angegeben ist.

Standardaktion: Die Task wird abnormal beendet.

50 NOPASSBKRD

Tritt auf, wenn kein Sparbuch vorhanden ist.

Standardaktion: Die Task wird abnormal beendet.

61 NOTALLOC

Tritt auf, falls die Anwendung nicht Eigner der im Befehl angegebenen Funktion ist.

Standardaktion: Die Task wird abnormal beendet.

02 RDATT

Tritt auf, falls ein Befehl RECEIVE mithilfe der Abruftaste anstelle der Rückföhrtaste beendet wird.

Standardaktion: Die Bedingung wird ignoriert.

81 TERMERR

Tritt für einen terminalbezogenen Fehler auf, wie zum Beispiel einen Sitzungsfehler. Diese Bedingung gilt nur für Terminals, die mit z/OS Communications Server verbunden sind.

Diese Bedingung kann durch eine Anforderung CANCEL TASK durch ein Knotenfehlerprogramm des Benutzers verursacht werden, falls für die Task eine Terminalsteuerungsanforderung aussteht, wenn der Sitzungsfehler durch das Programm des Knotens für abnormale Bedingungen abgewickelt wird.

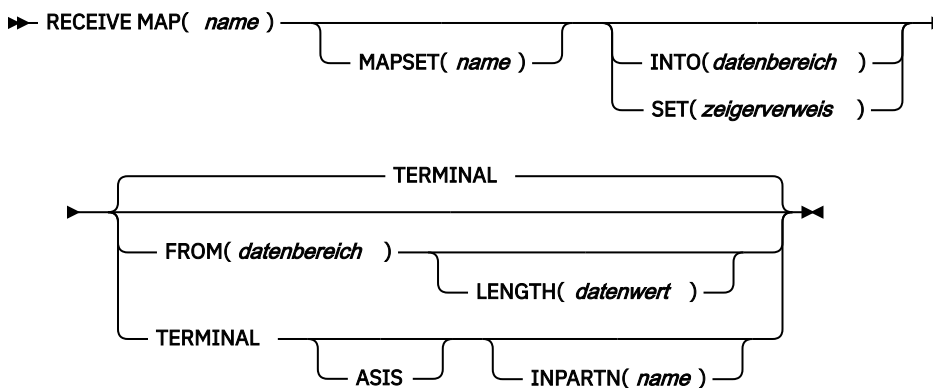
Standardaktion: Die Task wird abnormal mit dem Abbruchcode ATNI beendet.

RECEIVE MAP

Empfängt die Anzeigeeingabe in einem Anwendungsdatenbereich.

Weitere Informationen zu BMS finden Sie unter [Basic Mapping Support](#).

RECEIVE MAP



Bedingungen: EOC, EODS, INVMPSZ, INVPARTN, INVREQ, MAPFAIL, PARTNFAIL, RDATT, UNEXPIN

Anmerkung: Die Bedingung INPARTN wird bei standardmäßiger und bei vollständiger BMS unterstützt.

Beschreibung

Der Befehl RECEIVE MAP ordnet Eingabedaten aus einem Terminal in einem Datenbereich eines Anwendungsprogramms zu.

Daten aus bestimmten logischen Einheiten werden nicht zugeordnet, sondern bleiben unverändert. Im Handbuch für das entsprechende CICS-Subsystem erfahren Sie, ob dies für eine bestimmte logische Einheit zutrifft.

Im Anschluss an einen Befehl RECEIVE MAP wird die eingehende Cursorposition in EIBCPOSN platziert und das AID-Zeichen für das Terminal in EIBAID.

Maskendefinitionen finden Sie im Abschnitt BMS-Makros.

Falls Daten empfangen werden sollen, müssen Sie entweder die Option INTO oder die Option SET angeben. Wenn ein Befehl RECEIVE ausschließlich zu dem Zweck ausgegeben wird, ein AID-Zeichen zu erkennen, können Sie die Optionen INTO und SET weglassen.

Optionen

ASIS

Gibt an, dass Kleinbuchstaben im 3270-Eingabedatenstrom nicht in Großbuchstaben umgesetzt werden; hierdurch ist die aktuelle Task in der Lage, eine Nachricht zu empfangen, die Daten sowohl in Klein- als auch in Großschreibung enthält.

Auf den ersten Befehl RECEIVE einer Transaktion hat diese Option keinen Einfluss; gleiches gilt, falls die Anzeige Daten enthält, bevor eine Transaktion gestartet wird. Falls beispielsweise eine Transaktion durch eine andere Transaktion aufgerufen wird und als erstes Daten empfängt, die ursprünglich durch diese Transaktion ausgegeben wurden, kann die Umsetzung in Großbuchstaben für die Daten nicht unterdrückt werden. Diese Daten werden gelesen und zur Vorbereitung der nächsten Task umgesetzt. Der erste Befehl RECEIVE in dieser Task ruft dann die umgesetzten Daten ab.

Anmerkung: Falls Sie ein Katakana-Terminal verwenden, werden möglicherweise einige Nachrichten angezeigt, in denen englische und Katakana-Zeichen kombiniert sind. Dies liegt daran, dass Katakana-Terminals keine Ausgabe in Groß-/Kleinschreibung anzeigen können. Großbuchstaben im Datenstrom werden als englische Großbuchstaben angezeigt, Kleinbuchstaben jedoch als Katakana-Zeichen. Bitten Sie in einem solchen Fall Ihren Systemprogrammierer, in den Systeminitialisierungsparametern MSGCASE=UPPER anzugeben, damit diese Nachrichten ausschließlich Großbuchstaben enthalten.

FROM(datenbereich)

Gibt den Datenbereich an, in dem die Daten enthalten sind, die durch einen Befehl RECEIVE MAP zugeordnet werden sollen. Diese Angabe schließt das 12 Byte umfassende Präfix ein, das durch die Option TIOAPFX=YES für die BMS-Zuordnungsdefinitionen DFHMDI und DFHMSD generiert wird (siehe hierzu die Abschnitte über den DFHMDI-Operanden TERM und die Angabe von NODDS im BMS-Operanden).

INPARTN(name)

Gibt den 1 bis 2 Zeichen langen Namen der Partition an, in die der Terminalbediener voraussichtlich Daten eingeben wird. Falls der Terminalbediener Daten in einer anderen Partition eingibt, wird die für INPARTN angegebene Partition aktiviert, die Tastatur für die Partition entsperrt und eine Fehlernachricht an eine beliebige Fehlernachrichtenpartition ausgegeben. Falls das Terminal Partitionen nicht unterstützt oder es keine Anwendungspartitionsgruppe gibt, wird diese Option ignoriert.

INTO(datenbereich)

Gibt den Datenbereich an, in den die zugeordneten Daten geschrieben werden sollen. Wenn dieses Feld nicht angegeben ist, wird als Name standardmäßig der Name der Zuordnung mit dem Suffix I verwendet.

LENGTH(datenwert)

Gibt die Länge der zu formatierenden Daten in Form eines Halbwort-Binärwertes an. Der Wert darf die Länge des bei FROM angegebenen Datenbereichs nicht überschreiten, sollte jedoch die Länge des 12 Byte umfassenden Präfix einbeziehen, das durch die Option TIOAPFX=YES für die BMS-Zuordnungsdefinitionen DFHMDI und DFHMSD generiert wird (siehe hierzu die Abschnitte über den DFHMDI-Operanden TERM und die Angabe von NODDS im BMS-Operanden).

Eine Beschreibung einer sicheren Obergrenze finden Sie unter „LENGTH-Optionen in CICS-Befehlen“ auf Seite 10.

MAP(name)

Gibt den 1 bis 7 Zeichen langen Namen der zu verwendenden Zuordnung an.

MAPSET(name)

Gibt den 1 bis 7 Zeichen langen Namen ohne Suffix der zu verwendenden Maskengruppe an. Die Maskengruppe muss sich in der CICS-Programmbibliothek befinden. Sie kann entweder mithilfe der Onlineresourcendefinition (RDO) oder durch die automatische Programminstallation bei ihrer erstmaligen Verwendung definiert werden. Falls diese Option nicht angegeben wird, wird der in der Option MAP angegebene Name als Name der Maskengruppe angenommen.

SET(zeigerverweis)

Gibt den Zeiger an, der auf die Adresse des 12 Byte umfassenden Präfix für die zugeordneten Daten gesetzt werden soll. Das 12 Byte umfassende Präfix wird durch die Option TIOAPFX=YES für die BMS-Zuordnungsdefinitionen DFHMDI und DFHMSD generiert (siehe hierzu die Abschnitte über den DFHMDI-Operanden TERM und die Angabe von NODDS im BMS-Operanden).

Der Zeigerverweis ist bis zum nächsten Befehl RECEIVE oder bis zum Ende der Task gültig.

Falls dem Anwendungsprogramm DATALLOCATION(ANY) zugeordnet ist, kann die Adresse der Daten ober- oder unterhalb der 16-MB-Grenze liegen. Der Datenbereich schließt das 12 Byte lange Präfix ein. Achten Sie darauf, dass das 12 Byte umfassende Präfix nicht überschrieben wird, weil dies einen Fehler im Speicher verursachen könnte.

Wenn dem Anwendungsprogramm DATALLOCATION(BELOW) zugeordnet ist und sich die Daten oberhalb der 16-MB-Grenze befinden, werden die Daten an eine Adresse unterhalb der 16-MB-Grenze kopiert und die Adresse dieser Kopie wird zurückgegeben.

Wenn für die aktive Task TASKDATAKEY(USER) angegeben wird und der Speicherschutz aktiv ist, befinden sich die zurückgegebenen Daten in einem Benutzerschlüssel. Wenn für die aktive Task TASKDATAKEY(CICS) angegeben wird und der Speicherschutz aktiv ist, befinden sich die zurückgegebenen Daten in einem CICS-Schlüssel.

TERMINAL

Gibt an, dass die Eingabedaten aus dem Terminal gelesen werden sollen, von dem die Transaktion stammt.

Bedingungen

Einige der folgenden Bedingungen können kombiniert auftreten. Falls mehrere Bedingungen auftreten, wird nur die erste Bedingung an das Anwendungsprogramm übergeben.

EIBRCODE wird jedoch so festgelegt, dass alle aufgetretenen Bedingungen angegeben sind.

06 EOC

Tritt auf, falls die Anforderung/Antwort-Einheit mit festgelegtem Anzeiger für das Kettenende (End-of-chain, EOC) empfangen wird. Sie gilt nur für logische Einheiten.

Standardaktion: Die Bedingung wird ignoriert.

05 EODS

Tritt auf, wenn keine Daten empfangen werden (nur ein Funktionsverwaltungsheader). Diese Bedingung gilt nur für logische Einheiten mit 3770-Stapel und für logische Einheiten mit 3770- bzw. 3790-Stapeldatenaustausch.

Standardaktion: Die Task wird abnormal beendet.

38 INVMP SZ

Tritt auf, falls die angegebene Zuordnung zu breit oder zu lang für das Terminal ist.

Standardaktion: Die Task wird abnormal beendet.

65 INV PARTN

Tritt auf, falls die angegebene Partition nicht in der Partitionsgruppe definiert ist, die dem Anwendungsprogramm zugeordnet ist.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

Tritt auf, falls ein Befehl RECEIVE MAP in einer Task ausgegeben wird, die nicht für ein Terminal bestimmt ist; solche Tasks haben keinen Terminalein-/ausgabebereich (TIOA) oder Eintrag in der Terminalsteuertabelle (TCTTE).

Standardaktion: Die Task wird abnormal beendet.

36 MAPFAIL

Tritt auf, wenn die zuzuordnenden Daten die Länge Null besitzen oder keine Sequenz für das Festlegen der Pufferadresse (Set-buffer-address, SBA) enthalten. Sie gilt nur für logische 3270-Einheiten. Der empfangende Datenbereich enthält den nicht zugeordneten Eingabedatenstrom. Die Menge von nicht zugeordneten Daten, die in den Benutzerbereich verschoben werden, ist auf die Länge beschränkt, die mit der Option LENGTH angegeben ist. Die Eingabezuordnung wird nicht auf Nullen gesetzt.

Diese Bedingung tritt außerdem auf, falls ein Programm einen Befehl RECEIVE MAP ausgibt, auf den der Terminalbediener durch Drücken einer Löschtaste oder Programmabrufaste bzw. durch Drücken der Eingabetaste oder einer Funktionstaste ohne die Eingabe von Daten antwortet.

Standardaktion: Die Task wird abnormal beendet.

66 PARTNFAIL

Tritt auf, falls der Terminalbediener mehr als drei Mal versucht, Daten in einer anderen Partition einzugeben, als durch die Option INPARTN angegeben ist.

Standardaktion: Die Task wird abnormal beendet.

02 RDATT

Tritt auf, falls ein Befehl RECEIVE MAP durch den Bediener mithilfe der Abrufaste anstelle der Rückföhrtaste beendet wird. Sie gilt nur für das Kommunikationsterminal IBM 2741 und nur dann, wenn für CICS die Unterstützung des 2741-Leseabrufs generiert wurde.

Standardaktion: Die Bedingung wird ignoriert.

49 UNEXPIN

Tritt auf, wenn nicht erwartete oder nicht erkannte Daten empfangen werden. Diese Bedingung gilt nur für Stapeldatenaustauschterminals.

Standardaktion: Die Task wird abnormal beendet.

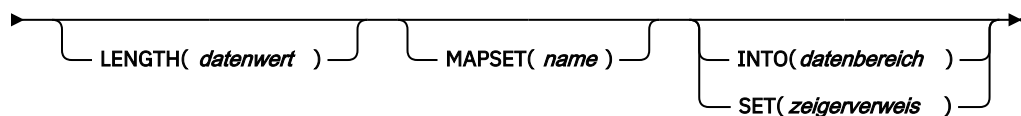
RECEIVE MAP MAPPINGDEV

Empfängt die Anzeigeeingabe in einem Anwendungsdatenbereich, ohne Verweis auf die Hauptfunktion, sofern vorhanden. Die Terminalmerkmale werden aus dem Parameter **MAPPINGDEV** abgerufen.

Weitere Informationen zu BMS finden Sie unter [Basic Mapping Support](#).

RECEIVE MAP MAPPINGDEV

➡ RECEIVE MAP(*name*) — MAPPINGDEV(*datenwert*) — FROM(*datenbereich*) —>



Bedingungen: INVMPsz, INVREQ, MAPFAIL

Beschreibung

Der Befehl RECEIVE MAP MAPPINGDEV ermöglicht die Zuordnung von Eingabedaten aus einem 3270-Terminal, das nicht notwendigerweise die Hauptfunktion der Transaktion darstellen muss.

MAPPINGDEV gibt den Namen eines 3270-Terminals an, dessen BMS-Merkmale zum Erstellen des Eingabedatenstroms verwendet wurden. Hierbei kann es sich um ein Terminal handeln, aus dem die Daten ursprünglich mit einem Befehl RECEIVE empfangen wurden.

Optionen

AID(datenwert)

Gibt den 1 Byte umfassenden Datenbereich an, der den Wert des 3270-AID-Zeichens enthält, der bei der Durchführung der Zuordnungsoperation verwendet werden soll. In der Regel ist dies der Wert, der im Feld EIBAID nach derjenigen Operation RECEIVE enthalten ist, die den Datenstrom ursprünglich vom Terminal empfangen hat.

Der angegebene Wert wird nach dem Abschluss der Operation in das Feld EIBAID im EXEC-Schnittstellenblock verschoben. Es findet keine Prüfung statt, ob der angegebene AID-Wert gültig ist.

Falls die Option AID(datenwert) nicht angegeben ist, wird standardmäßig X'7D' (also die Eingabetaste) als AID-Wert verwendet.

Falls das AID-Byte (entweder explizit oder standardmäßig) eine andere Operation als CLEAR, PA1, PA2 oder PA3 angibt und für die Zuordnung die Einstellung CURSLOC=YES angegeben ist, wird das Feld, das den Cursor enthält, durch eine Festlegung des Bit X'02' in seinem Markierungsbyte markiert.

Falls der AID-Wert (entweder explizit angegeben oder standardmäßig) Gegenstand eines Befehls HANDLE AID ist, wird die angegebene Verzweigung auf die übliche Weise verwendet.

CURSOR(datenwert)

Gibt ein Feld mit einer Halbwort-Binärzahl ohne Vorzeichen an, das die zu verwendende Cursorposition (bezogen auf Null) enthält. In der Regel ist dies der Wert, der im Feld EIBCPOSN nach derjenigen Operation RECEIVE enthalten ist, die den Datenstrom ursprünglich vom Terminal empfangen hat.

Der angegebene Wert wird nach dem Abschluss der Operation in das Feld EIBAID im EXEC-Schnittstellenblock verschoben. Es findet keine Prüfung statt, ob der angegebene Wert für CURSOR gültig ist.

Falls die Option CURSOR(datenwert) nicht angegeben ist, wird standardmäßig X'0000' (also die Eingabetaste) als Cursorwert verwendet.

FROM(datenbereich)

Gibt den Datenbereich an, in dem die Daten enthalten sind, die zugeordnet werden sollen. Diese Angabe muss das Format eines Terminalein-/ausgabebereichs besitzen und ein 12 Byte umfassendes Präfix enthalten.

INTO(datenbereich)

Gibt den Datenbereich an, in den die zugeordneten Daten geschrieben werden sollen. Wenn dieses Feld nicht angegeben ist, wird als Name standardmäßig der Name der Zuordnung mit dem Suffix I verwendet.

LENGTH(datenwert)

Gibt die Länge der zu formatierenden Daten in Form eines Halbwort-Binärwertes an. Der Wert darf die Länge des bei FROM angegebenen Datenbereichs nicht überschreiten, sollte jedoch die Länge des 12 Byte umfassenden Präfix einbeziehen, das durch die Option TIOAPFX=YES für die BMS-Zuordnungsdefinitionen DFHMDI und DFHMSD generiert wird (siehe hierzu die Abschnitte über den DFHMDI-Operanden TERM und die Angabe von NODDS im BMS-Operanden). Eine Beschreibung einer sicheren Obergrenze finden Sie unter „LENGTH-Optionen in CICS-Befehlen“ auf Seite 10.

MAP(name)

Gibt den 1 bis 7 Zeichen langen Namen der zu verwendenden Zuordnung an.

MAPPINGDEV(datenwert)

Gibt den Namen eines 3270-Terminals an, dessen Merkmale mit denen des Terminals übereinstimmen, aus dem die Daten ursprünglich mit einem Befehl RECEIVE empfangen wurden.

MAPSET(name)

Gibt den 1 bis 7 Zeichen langen Namen ohne Suffix der zu verwendenden Maskengruppe an. Die Maskengruppe muss sich in der CICS-Programmbibliothek befinden. Sie kann entweder mithilfe der On-

linere sourcedefinition (RDO) oder durch die automatische Programminstallation bei ihrer erstmaligen Verwendung definiert werden. Falls diese Option nicht angegeben wird, wird der in der Option MAP angegebene Name als Name der Maskengruppe angenommen.

SET(zeigerverweis)

Gibt den Zeiger an, der auf die Adresse des 12 Byte umfassenden Präfix für die zugeordneten Daten gesetzt werden soll. Der Zeigerverweis ist bis zum nächsten Befehl RECEIVE oder RECEIVE MAP bzw. bis zum Ende der Transaktion gültig, sofern er nicht durch die Anwendung mit einer Anforderung FREEMAIN freigegeben wird.

Falls für die aktive Task TASKDATALOC(ANY) angegeben ist, können sich die zurückgegebenen Daten ober- oder unterhalb der 16-MB-Grenze befinden.

Falls für die aktive Task TASKDATALOC(BELOW) angegeben ist, befinden sich die zurückgegebenen Daten unterhalb der 16-MB-Grenze.

Falls für die aktive Task TASKDATAKEY(USER) angegeben und der Speicherschutz aktiv ist, befinden sich die zurückgegebenen Daten im Benutzerschlüssel. Wenn für die aktive Task TASKDATAKEY(CICS) angegeben wird und der Speicherschutz aktiv ist, befinden sich die zurückgegebenen Daten in einem CICS-Schlüssel.

Bedingungen

Einige der folgenden Bedingungen können kombiniert auftreten. Falls mehrere Bedingungen auftreten, wird nur die erste Bedingung an das Anwendungsprogramm übergeben.

38 INVMP SZ

Tritt auf, falls die angegebene Zuordnung zu breit oder zu lang für das Terminal ist.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

Tritt auf, falls das durch MAPPINGDEV angegebene Terminal nicht vorhanden ist, BMS nicht unterstützt oder kein 3270-Drucker bzw. keine 3270-Anzeige ist.

Standardaktion: Die Task wird abnormal beendet.

36 MAPFAIL

Tritt auf, wenn die zuzuordnenden Daten die Länge Null besitzen oder keine Sequenz für das Festlegen der Pufferadresse (Set-buffer-address, SBA) enthalten.

Standardaktion: Die Task wird abnormal beendet.

RECEIVE PARTN

Empfängt Daten von einer 8775-Terminalpartition. Dieser Befehl ist nur bei standardmäßiger und vollständiger BMS verfügbar.

Weitere Informationen zu BMS finden Sie unter [Basic Mapping Support](#).

RECEIVE PARTN

➡➡ RECEIVE PARTN(*datenbereich*) ➡➡

Standardmäßige und vollständige BMS:

RECEIVE PARTN

➡➡ INTO(*datenbereich*) ——— LENGTH(*datenwert*) ——— ➡➡
 SET(*zeigerverweis*) ——— ASIS ———

Bedingungen: EOC, EODS, INVPARTN, INVREQ, LENGERR

Beschreibung

Der Befehl RECEIVE PARTN liest Daten aus einer Partition auf einem 8775-Terminal. Er gibt an, aus welcher Partition die Daten stammen, und stellt die Daten in den mit INTO oder SET angegebenen Datenbereich. Anschließend können Sie die Daten so verarbeiten, als ob sie aus einem Terminal im Basisstatus (also im unpartitionierten Zustand) stammen würden.

Im Anschluss an einen Befehl RECEIVE PARTN wird die eingehende Cursorposition im Feld EIBCPOSN platziert und das AID-Zeichen für das Terminal im Feld EIBAID. Die Felder EIBAID und EIBCPOSN werden außerdem bei der Taskinitialisierung von Nicht-ATI-Tasks sowie nach jeder Terminalsteuerung und BMS-Eingabe aktualisiert.

Maskendefinitionen finden Sie im Abschnitt [BMS-Makros](#).

Falls Daten empfangen werden sollen, müssen Sie entweder die Option INTO oder die Option SET angeben. Wenn ein Befehl RECEIVE ausschließlich zu dem Zweck ausgegeben wird, ein AID-Zeichen zu erkennen, können Sie die Optionen INTO und SET weglassen.

Optionen

ASIS

Gibt an, dass Kleinbuchstaben im 3270-Eingabedatenstrom nicht in Großbuchstaben umgesetzt werden; hierdurch ist die aktuelle Task in der Lage, eine Nachricht zu empfangen, die Daten sowohl in Klein- als auch in Großschreibung enthält.

Auf den ersten Befehl RECEIVE einer Transaktion hat die Option ASIS keinen Einfluss; gleiches gilt, falls die Anzeige Daten enthält, bevor eine Transaktion gestartet wird. Falls beispielsweise eine Transaktion durch eine andere Transaktion aufgerufen wird und als erstes Daten empfängt, die ursprünglich durch diese Transaktion ausgegeben wurden, kann die Umsetzung in Großbuchstaben für die Daten nicht unterdrückt werden. Diese Daten werden gelesen und zur Vorbereitung der nächsten Task umgesetzt. Der erste Befehl RECEIVE in dieser Task ruft dann die umgesetzten Daten ab.

Anmerkung: Falls Sie ein Katakana-Terminal verwenden, werden möglicherweise einige Nachrichten angezeigt, in denen englische und Katakana-Zeichen kombiniert sind. Dies liegt daran, dass Katakana-Terminals keine Ausgabe in Groß-/Kleinschreibung anzeigen können. Großbuchstaben im Datenstrom werden als englische Großbuchstaben angezeigt, Kleinbuchstaben jedoch als Katakana-Zeichen. Bitten Sie in einem solchen Fall Ihren Systemprogrammierer, in den Systeminitialisierungsparametern MSGCASE=UPPER anzugeben, damit diese Nachrichten ausschließlich Großbuchstaben enthalten.

INTO(datenbereich)

Gibt den Bereich an, in den die aus den Partitionssteuerungen stammenden Daten geschrieben werden sollen. Die Länge dieses Bereichs muss durch die Option LENGTH angegeben sein. Falls der Bereich nicht groß genug für die Aufnahme der Eingabedaten ist, werden die Eingabedaten abgeschnitten und die Bedingung LENGERR tritt auf. Der in der Option LENGTH angegebene Datenbereich wird auf die Länge der empfangenen Daten vor dem Abschneiden gesetzt.

LENGTH(datenwert)

Gibt die Länge der zu formatierenden Daten in Form eines Halbwort-Binärwertes an. Als Wert muss die Länge eines beliebigen Bereichs von INTO vor dem Befehl festgelegt sein. Nach dem Befehl legt BMS für die Option LENGTH die Länge der empfangenen Daten vor dem Abschneiden fest, falls der für INTO angegebene Bereich zu klein ist.

Eine Beschreibung einer sicheren Obergrenze finden Sie unter [„LENGTH-Optionen in CICS-Befehlen“](#) auf Seite 10.

PARTN(datenbereich)

Wird auf den 1 bis 2 Zeichen langen Namen der Eingabepartition gesetzt. Die Partition kann entweder mithilfe der Onlinere Ressourcendefinition (RDO) oder durch die automatische Programminstallation bei ihrer erstmaligen Verwendung definiert werden.

SET(zeigerverweis)

Gibt den Zeiger an, der auf die Adresse des 12 Byte umfassenden Präfix für die zugeordneten Daten gesetzt werden soll. Der Zeigerverweis ist bis zum nächsten Befehl RECEIVE oder bis zum Ende der Task gültig.

Falls dem Anwendungsprogramm DATALOCATION(ANY) zugeordnet ist, kann die Adresse der Daten ober- oder unterhalb der 16-MB-Grenze liegen.

Wenn dem Anwendungsprogramm DATALOCATION(BELOW) zugeordnet ist und sich die Daten oberhalb der 16-MB-Grenze befinden, werden die Daten an eine Adresse unterhalb der 16-MB-Grenze kopiert und die Adresse dieser Kopie wird zurückgegeben.

Wenn für die aktive Task TASKDATAKEY(USER) angegeben wird und der Speicherschutz aktiv ist, befinden sich die zurückgegebenen Daten in einem Benutzerschlüssel. Wenn für die aktive Task TASKDATAKEY(CICS) angegeben wird und der Speicherschutz aktiv ist, befinden sich die zurückgegebenen Daten in einem CICS-Schlüssel.

Bedingungen

Einige der folgenden Bedingungen können kombiniert auftreten. Falls mehrere Bedingungen auftreten, wird nur die erste Bedingung an das Anwendungsprogramm übergeben.

06 EOC

Tritt auf, falls die Anforderung/Antwort-Einheit mit festgelegtem Anzeiger für das Kettenende (End-of-chain, EOC) empfangen wird. Sie gilt nur für logische Einheiten.

Standardaktion: Die Bedingung wird ignoriert.

05 EODS

Tritt auf, wenn keine Daten empfangen werden (nur ein Funktionsverwaltungsheader). Diese Bedingung gilt nur für logische Einheiten mit 3770-Stapel und für logische Einheiten mit 3770- bzw. 3790-Stapeldatenaustausch.

Standardaktion: Die Task wird abnormal beendet.

65 INVPARTN

Tritt auf, falls die angegebene Partition nicht in der Partitionsgruppe definiert ist, die dem Anwendungsprogramm zugeordnet ist.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

Tritt auf, falls ein Befehl RECEIVE PARTN in einer Task ausgegeben wird, die nicht für ein Terminal bestimmt ist; solche Tasks haben keinen Terminalein-/ausgabebereich (TIOA) oder Eintrag in der Terminalsteuertabelle (TCTTE).

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

Tritt auf, falls der für die Option INTO eines Befehls RECEIVE PARTN angegebene Bereich nicht groß genug für die Aufnahme der Eingabedaten ist.

Standardaktion: Die Daten werden so abgeschnitten, dass sie in den für INTO angegebenen Bereich passen.

RELEASE

Gibt ein geladenes Programm, eine geladene Tabelle oder eine geladene Maskengruppe frei.

RELEASE

➡ RELEASE — PROGRAM(*name*) →

Bedingungen: INVREQ, NOTAUTH, PGMIDERR

Dieser Befehl ist threadsicher.

Hinweis für dynamisches Transaktionsrouting: Die Verwendung des Befehls RELEASE für ein Programm, das mit der Option HOLD geladen wurde, kann transaktionsübergreifende Affinitäten erzeugen, die die Nutzung des dynamischen Transaktionsroutings beeinträchtigen. Weitere Informationen zu Transaktionsaffinitäten finden Sie unter [Affinität](#).

Beschreibung

Der Befehl RELEASE gibt das Programm, die Tabelle oder die Maskengruppe frei, das/die zuvor durch einen Befehl LOAD geladen wurde. Dies hat zur Folge, dass die ausgebende Task die Ressource erst nach Ausgabe eines weiteren Befehls LOAD wieder verwenden kann.

Anmerkung: Der Befehl RELEASE entfernt nicht ein Programm aus dem Speicher. Er verringert den Wert von RESCOUNT um 1; sobald der Zähler den Wert Null erreicht, kann der durch das Programm belegte Speicher durch den CICS-Speichermanager freigegeben werden.

Falls die Option HOLD im Befehl LOAD angegeben ist, wird die geladene Ressource am Ende der Task nicht freigegeben. Die Freigabe der Ressource ist nur durch einen Befehl RELEASE möglich. Dieser Befehl RELEASE kann durch die Task ausgegeben werden, die die Ressource geladen hat, aber auch durch jede andere Task.

Falls die Option HOLD nicht im Befehl LOAD angegeben ist, wird die geladene Ressource am Ende der Task freigegeben. Sie kann jedoch durch die Task, die die Ressource geladen hat, auch zuvor durch Ausgabe eines Befehls RELEASE freigegeben werden.

Optionen

PROGRAM(name)

Gibt die Kennung (1-8 Zeichen) eines Programms, einer Tabelle oder einer Maskengruppe an, das/die freigegeben werden soll.

Bedingungen

16 INVREQ

RESP2-Werte:

5

Durch das Programm wurde ungültigerweise versucht, sich selbst freizugeben. Ein Befehl RELEASE für das Programm, das diesen Befehl enthält, ist nur dann zulässig, wenn von derselben Task ein korrespondierender Befehl LOAD für das Programm ausgegeben wurde oder wenn von einer anderen Task ein Befehl LOAD mit der Option HOLD ausgegeben wurde.

6

Der Befehl wurde für ein Programm ausgegeben, das nicht geladen ist.

7

Entweder wurde der Befehl für ein Programm ausgegeben, das von einer anderen Task ohne die Option HOLD geladen wurde, oder das Programm wurde als globaler Benutzerexit aktiviert.

17

Das Programm ist mit der Einstellung RELOAD=YES definiert. Es muss durch einen Befehl FREE-MAIN anstelle eines Befehls RELEASE freigegeben werden.

30

Die Programmmanagerdomäne wurde noch nicht initialisiert. Dies liegt wahrscheinlich daran, dass eine Freigabeanforderung in einem PLT-Programm der ersten Phase ausgegeben wurde.

Standardaktion: Die Task wird abnormal beendet.

70 NOTAUTH

Tritt auf, wenn eine Ressourcensicherheitsprüfung für PROGRAM(name) fehlgeschlagen ist.

Standardaktion: Die Task wird abnormal beendet.

27 PGMIDERR

RESP2-Werte:

- 1** Für ein Programm, eine Tabelle oder eine Maskengruppe ist keine Ressourcendefinition installiert.
- 2** Ein Programm, eine Tabelle oder eine Maskengruppe ist inaktiviert.
- 9** Die installierte Programmdefinition bezieht sich auf ein fernes Programm.
- 42** Es wurde versucht, den Befehl RELEASE für ein JVM-Programm auszuführen. Dies ist ungültig, weil Programme mit Java-Bytecode nicht durch das CICS-Ladeprogramm verwaltet werden.

Standardaktion: Die Task wird abnormal beendet.

Beispiele

Das folgende Beispiel zeigt, wie ein Anwendungsprogramm namens PROG4 freigegeben wird, das infolge eines Befehls LOAD geladen wurde:

```
EXEC CICS RELEASE PROGRAM('PROG4')
```

REMOVE SUBEVENT

Entfernt ein Unterereignis aus einem zusammengesetzten BTS-Ereignis.

REMOVE SUBEVENT

➤ REMOVE — SUBEVENT(*datenwert*) — EVENT(*datenwert*) ➤

Bedingungen: EVENTERR, INVREQ

Beschreibung

Der Befehl REMOVE SUBEVENT entfernt ein Unterereignis aus einem angegebenen zusammengesetzten BTS-Ereignis.

Durch diesen Aufruf wird das entfernte Ereignis nicht gelöscht. Es wird auch nicht auf den Auslösestatus des Ereignisses zurückgesetzt. Beachten Sie, dass nach diesem Aufruf das entfernte Ereignis, da es kein Unterereignis mehr ist, wenn es ausgelöst wird, bewirkt, dass die aktuelle Aktivität erneut aktiviert (angeschlossen) wird.

Das Entfernen eines Unterereignisses hat zur Folge, dass das Prädikat des zusammengesetzten Ereignisses erneut ausgewertet wird.

Optionen

EVENT(datenwert)

Gibt den Namen (1 - 16 Zeichen) des zusammengesetzten Ereignisses an.

SUBEVENT(datenwert)

Gibt den Namen (1 - 16 Zeichen) des Ereignisses an, das aus dem angegebenen zusammengesetzten Ereignis entfernt werden soll.

Bedingungen

111 EVENTERR

RESP2-Werte:

4

Das in der Option EVENT angegebene Ereignis wird von BTS nicht erkannt.

5

Das in der Option SUBEVENT angegebene Unterereignis wird von BTS nicht erkannt.

16 INVREQ

RESP2-Werte:

1

Der Befehl wurde außerhalb des Gültigkeitsbereichs einer Aktivität abgesetzt.

2

Das in der Option EVENT angegebene Ereignis ist kein zusammengesetztes Ereignis.

3

Das in der Option SUBEVENT angegebene Ereignis ist kein Unterereignis des zusammengesetzten Ereignisses, das in der Option EVENT angegeben wurde.

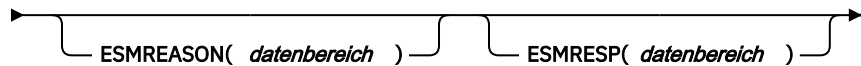
REQUEST ENCRYPTPTKT

Fordert bei einem externen Sicherheitsmanager die Generierung eines verschlüsselten Passtickets an.

REQUEST ENCRYPTPTKT

➡ REQUEST — ENCRYPTPTKT(*zeigerverweis*) — FLENGTH(*datenbereich*) →

↳ ENCRYPTKEY(*datenbereich*) — ESMAPPNAME(*datenwert*) →



Bedingungen: NOTAUTH, INVREQ

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl **REQUEST ENCRYPTPTKT** fordert bei einem externen Sicherheitsmanager wie beispielsweise RACF an, dass ein verschlüsseltes Passticket generiert wird.

Ein nicht verschlüsseltes Passticket ist ein Kennwortersatz, mit dem sich Ihr Programm bei einer bestimmten Anwendung auf einem bestimmten System anmelden kann, z. B. bei einer anderen CICS-Region. Mit der Option ESMAPPNAME müssen Sie den Profilnamen angeben, durch den der externe Sicherheitsmanager die Anwendung referenziert, bei der Sie sich anmelden.

Die Option ENCRYPTKEY muss einen Wert enthalten, der durch einen vorherigen Befehl **VERIFY TOKEN** für dieselbe CICS-Task zurückgegeben wurde.

Anmerkung: Wenn ein Befehl **VERIFY TOKEN** mit der Option ENCRYPTKEY ausgeführt wird, ist jeder vorherige Wert von ENCRYPTKEY, der für dieselbe Task abgerufen wurde, nicht mehr gültig. Dies bedeutet, dass bei einer Ausführung von mehreren Befehlen **VERIFY TOKEN** lediglich der Wert von ENCRYPTKEY aus dem letzten Befehl für die Verwendung mit einem Befehl **REQUEST ENCRYPTPTKT** gültig ist.

Ein Wert in ENCRYPTKEY kann ein Mal verwendet werden, um ein verschlüsseltes Passticket mit dem Befehl **REQUEST ENCRYPTPTKT** abzurufen. Falls ein weiteres verschlüsseltes Passticket erforderlich ist, nachdem ein Befehl **REQUEST ENCRYPTPTKT** ausgeführt wurde, muss erneut ein Befehl **VERIFY TOKEN** ausgeführt werden, um einen neuen Wert für die Option ENCRYPTKEY anzufordern. Anschließend können Sie ein zweites Mal einen Befehl **REQUEST ENCRYPTPTKT** ausführen.

Das verschlüsselte Passticket, das durch einen Befehl **REQUEST ENCRYPTPTKT** zurückgegeben wird, ist zur Entschlüsselung durch das Subsystem gedacht, von dem das Kerberos-Token bereitgestellt wurde, das beim Befehl **VERIFY TOKEN** mit der Option ENCRYPTKEY verwendet wurde.

Das vom externen Sicherheitsmanager generierte verschlüsselte Passticket ist für die Benutzer-ID bestimmt, die der Task zugeordnet ist, von der der Befehl **REQUEST ENCRYPTPTKT** ausgegeben wird. Mit dem Befehl **EXEC CICS ASSIGN** und der Option **USERID** können Sie die Benutzer-ID ermitteln, die der Task zugeordnet ist. Für die Standard-Benutzer-ID kann kein Passticket angefordert werden.

Anmerkung: Eine Anforderung für ein Passticket ist erfolgreich, falls eine Benutzer-ID entzogen wird, jedoch ein Versuch fehlschlägt, sich mit dieser Benutzer-ID und einem Passticket anzumelden.

Ein Passticket muss innerhalb von 10 Minuten nach seiner Generierung genutzt werden. Falls dieses Zeitlimit für das Passticket überschritten wird (beispielsweise aufgrund eines Sitzungsfehlers), muss Ihre Anwendung ein weiteres Passticket generieren, bevor Sie erneut versuchen können, sich anzumelden. Wiederholt fehlgeschlagene Anmeldeversuche mit Passtickets können dazu führen, dass die Benutzer-ID entzogen wird.

Vor der Verwendung von Passtickets müssen Sie sicherstellen, dass die Systemuhren für das Zielsystem und das Ursprungssystem innerhalb des gültigen Zeitbereichs synchronisiert sind. Außerdem müssen Sie für jedes Zielsystem einen Schlüssel für die gesicherte Anmeldung definieren. Weitere Informationen zu den Voraussetzungen der Verwendung von Passtickets finden Sie unter [Passtickets für gesicherte Anmeldung generieren und verwenden](#).

Optionen

ENCRYPTPTKT(zeigerverweis)

Gibt einen Datenbereich an, in dem das verschlüsselte Passticket zurückgegeben wird.

FLENGTH(datenbereich_als_vollwort-binärzahl)

Gibt die Länge des verschlüsselten Passtickets zurück.

ENCRYPTKEY(datenbereich)

Ein vier Byte langes Token, das durch einen vorherigen Befehl **VERIFY TOKEN** abgerufen wurde und den Schlüssel für die Verschlüsselung des Passtickets darstellt.

ESMAPNAME(datenwert)

Gibt den acht Zeichen langen Profilnamen an, durch den der externe Sicherheitsmanager die Anwendung referenziert, für die das bereitgestellte Passticket verwendet wird. Handelt es sich bei der Anwendung beispielsweise um eine andere CICS-Region, könnte der Profilname die Anwendungs-ID (APPLID) der CICS-Region sein. Falls Sie als externen Sicherheitsmanager RACF verwenden, finden Sie weitere Informationen zu RACF-Profilnamen und Passtickets unter ['Using the secured signon function'](#) in ['z/OS Security Server RACF Security Administrator's Guide'](#).

ESMRESP(datenbereich)

Gibt den Antwortcode aus dem externen Sicherheitsmanager als Vollwort-Binärzahlvariable zurück. Bei Verwendung von RACF handelt es sich bei den möglichen Werten um die folgenden Rückkehrcodes, die zurückgegeben werden, wenn Sie den RACF-Service für die gesicherte Anmeldung zur Generierung eines Passtickets verwenden:

0

Ein Passticket wurde erstellt.

8

Es wurde kein Passticket erstellt.

ESMREASON(datenbereich)

Gibt den Ursachencode aus dem externen Sicherheitsmanager als Vollwort-Binärzahlvariable zurück.

Falls Sie RACF als externen Sicherheitsmanager verwenden, handelt es sich bei den möglichen Werten um die RACF-Ursachencodes, die nach Auswahl der Optionen **z/OS Security Server RACF Callable Services > GSS-API functions (Function code 2) > Subfunction codes > Wrap a message** zurückgegeben werden. Weitere Informationen finden Sie unter [GSS-API-Unterfunktionscodes in z/OS Security Server RACF Callable Services](#). Erläuterungen für Ursachencodes enthalten die Abschnitte [R_GenSec-Rückgabecodes](#) und [-Ursachencodes in z/OS Security Server RACF Callable Services](#) und [Statuscodes in z/OS Integrated Security Services Network Authentication Service Administration](#).

Bedingungen

16 INVREQ

RESP2-Werte:

251

Die Schnittstelle zwischen CICS und dem externen Sicherheitsmanager ist nicht aktiv.

252

Der vom externen Sicherheitsmanager zurückgegebene Wert in der Option ESMRESP ist nicht von CICS klassifiziert. Eine Erläuterung der Optionen ESMRESP und ESMREASON finden Sie in der ESM-Dokumentation.

254

Der externe Sicherheitsmanager unterstützt keine Anforderungen für ein Passticket.

255

Der Wert für ENCRYPTKEY ist ungültig.

256

Dieser Befehl ist nicht gültig, wenn Sie ihn unter der Standard-Benutzer-ID ausführen.

257

Die Option ENCRYPTTOKEN wurde zwar angegeben, aber das zugehörige Kerberos-Token stammt aus einem System, das den Nachrichtenschutz nicht unterstützt.

70 NOTAUTH

RESP2-Werte:

250

Das Passticket wurde nicht erfolgreich erstellt. Der externe Sicherheitsmanager autorisiert keine Anforderung nach einem Passticket für die Kombination aus der Benutzer-ID, die der Task zugeordnet ist, von der dieser Befehl ausgegeben wurde, mit dem Profilnamen, der in ESMAPPNAME angegeben ist.

260

Der externe Sicherheitsmanager autorisiert keine Anforderung nach der Generierung eines Passtickets für diese Region.

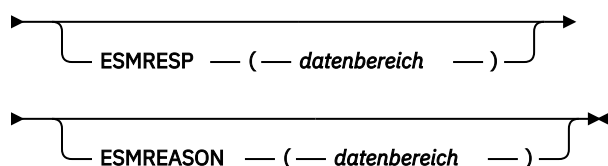
REQUEST PASSTICKET

Fordert die Erstellung eines Passtickets von einem externen Sicherheitsmanager an.

Syntax

REQUEST PASSTICKET

➤ REQUEST PASSTICKET — (— *data-area* —) — ESMAPPNAME — (— *datenbereich* —) ➔



Bedingungen: NOTAUTH, INVREQ

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl **REQUEST PASSTICKET** fordert bei einem externen Sicherheitsmanager wie beispielsweise RACF die Erstellung eines Passtickets an. Das Passticket ist ein Kennwortersatz, mit dem sich Ihr Programm bei einer bestimmten Anwendung auf einem bestimmten System anmelden kann, z. B. bei einer

anderen CICS-Region. Mit der Option ESMAPPNAME müssen Sie den Profilnamen angeben, durch den der externe Sicherheitsmanager die Anwendung referenziert, bei der Sie sich anmelden.

Das vom externen Sicherheitsmanager generierte Passticket ist für die Benutzer-ID bestimmt, die der Task zugeordnet ist, von der der Befehl **REQUEST PASSTICKET** ausgegeben wird. Mit dem Befehl **EXEC CICS ASSIGN** und der Option USERID können Sie die Benutzer-ID ermitteln, die der Task zugeordnet ist.

Anmerkung: Eine Anforderung für ein Passticket ist erfolgreich, falls eine Benutzer-ID entzogen wird, jedoch ein Versuch fehlschlägt, sich mit dieser Benutzer-ID und einem Passticket anzumelden.

Ein Passticket muss innerhalb von 10 Minuten nach seiner Generierung genutzt werden. Falls dieses Zeitlimit für das Passticket überschritten wird (beispielsweise aufgrund eines Sitzungsfehlers), muss Ihre Anwendung ein weiteres Passticket generieren, bevor Sie erneut versuchen können, sich anzumelden. Wiederholt fehlgeschlagene Anmeldeversuche mit Passtickets können dazu führen, dass die Benutzer-ID entzogen wird.

Bei Verwendung der CICS-Funktion 'Execution Diagnostic Facility' (EDF) werden Passtickets nicht angezeigt.

Vor der Verwendung von Passtickets müssen Sie sicherstellen, dass die Systemuhren für das Zielsystem und das Ursprungssystem innerhalb des gültigen Zeitbereichs synchronisiert sind. Außerdem müssen Sie für jedes Zielsystem einen Schlüssel für die gesicherte Anmeldung definieren. Weitere Informationen zu den Voraussetzungen der Verwendung von Passtickets finden Sie unter [Passtickets für gesicherte Anmeldung generieren und verwenden](#).

Optionen

ESMAPPNAME(datenbereich)

Gibt den acht Zeichen langen Profilnamen an, durch den der externe Sicherheitsmanager die Anwendung referenziert, für die das bereitgestellte Passticket verwendet wird. Bei CICS-Regionen ist der Profilname die Anwendungs-ID (APPLID) der CICS-Region. Falls Sie als externen Sicherheitsmanager RACF verwenden, finden Sie weitere Informationen zu RACF-Profilnamen und Passtickets unter ['Using the secured signon function' in 'z/OS Security Server RACF Security Administrator's Guide'](#).

ESMREASON(datenbereich)

Zeigt den für die ESM-Funktion zurückgegebenen ESM-Ursachencode an. Der Code wird zurückgegeben, wenn CICS eine Antwort NOTAUTH RESP zurückgibt. Details des ESM-Ursachencodes finden Sie unter den zugehörigen RESP2-Werten.

ESMRESP(datenbereich)

Zeigt den für die ESM-Funktion zurückgegebenen ESM-Rückgabecode an. Der Code wird zurückgegeben, wenn CICS eine Antwort NOTAUTH RESP zurückgibt. Details des ESM-Rückgabecodes finden Sie unter den zugehörigen RESP2-Werten.

PASSTICKET(datenbereich)

Gibt das acht Zeichen lange Passticket zurück, das durch den externen Sicherheitsmanager generiert wurde.

Bedingungen

16 INVREQ

RESP2-Werte:

247

Für ESMAPPNAME wurde ein ungültiger Wert angegeben.

251

Die Schnittstelle zwischen CICS und dem externen Sicherheitsmanager ist nicht aktiv.

252

Der vom externen Sicherheitsmanager zurückgegebene Wert in der Option ESMRESP ist nicht von CICS klassifiziert. Eine Erläuterung der Optionen ESMRESP und ESMREASON finden Sie in der ESM-Dokumentation.

254

Der externe Sicherheitsmanager unterstützt keine Anforderungen für ein Passticket.

256

Dieser Befehl ist nicht gültig, wenn er unter der Standard-Benutzer-ID ausgeführt wird.

70 NOTAUTH

RESP2-Werte:

250

Der externe Sicherheitsmanager (ESM) autorisiert keine Anforderung nach einem Passticket für die Kombination aus der Benutzer-ID, die der Task zugeordnet ist, von der dieser Befehl ausgegeben wurde, mit dem Profilnamen, der in ESMAPPNAME angegeben ist.

Wenn Sie RACF verwenden, finden Sie Informationen zu den zum Generieren von Passtickets erforderlichen RACF-Definitionen in den Anweisungen für das Zielsystem, die unter [Passtickets für die gesicherte Anmeldung implementieren](#) beschrieben werden.

Wenn RACF als externer Sicherheitsmanager verwendet wird, sind die Rückgabecodes und Ursachencodes in der Funktion **RACROUTE REQUEST=AUTH** definiert. Weitere Informationen enthält der Abschnitt [z/OS Security Server RACROUTE Macro Reference: Return codes and reason codes for RACROUTE REQUEST=AUTH \(standard form\)](#).

260

Der externe Sicherheitsmanager (ESM) autorisiert keine Anforderung nach einem Passticket für die Kombination aus der Benutzer-ID, die der Task zugeordnet ist, von der dieser Befehl ausgegeben wurde, mit dem Profilnamen, der in ESMAPPNAME angegeben ist.

Wenn Sie RACF verwenden, finden Sie Informationen zu den zum Generieren von Passtickets erforderlichen RACF-Definitionen in den Anweisungen für das Ursprungssystem, die unter [Passtickets für die gesicherte Anmeldung implementieren](#) beschrieben werden.

Wenn RACF als externer Sicherheitsmanager verwendet wird, sind die Rückgabecodes und Ursachencodes im RACF-Service für geschützte Passtickets zur Anmeldung (RCVTPTGN) beschrieben. Weitere Informationen enthält der Abschnitt [Using the RACF secured signon PassTicket service to generate a PassTicket](#).

RESET ACQPROCESS

Setzt einen BTS-Prozess auf seinen Anfangsstatus zurück.

RESET ACQPROCESS

➡ RESET — ACQPROCESS ➡

Bedingungen: INVREQ, IOERR, LOCKED, NOTAUTH, PROCESSBUSY, PROCESSERR

Beschreibung

Der Befehl RESET ACQPROCESS setzt den momentan abgerufenen BTS-Prozess auf seinen Anfangsstatus zurück. Alle untergeordneten Aktivitäten der Stammaktivität werden gelöscht.

Anmerkung: Der Befehl RESET hat keine Auswirkung auf die Prozesscontainer und die Container der Stammaktivität, deren Inhalt unverändert bleibt.

Setzen Sie diesen Befehl vor einem zweiten Befehl RUN ab, wenn ein Prozess erneut versucht werden muss. Wenn der Prozess erneut ausgeführt wird, wird der Stammaktivität ein Ereignis DFHINITIAL gesendet.

Für einen Prozess muss Folgendes zutreffen, damit er zurückgesetzt werden kann:

1. Er muss in der aktuellen Arbeitseinheit (UOW) abgerufen worden sein, das heißt, er muss der momentan angeforderte Prozess sein.

2. Er muss einen der folgenden Modi haben:

- COMPLETE. Dies ist der Normalfall. Möglicherweise wurde der Prozess abnormal beendet und muss zurückgesetzt werden, bevor er erneut versucht wird.
- INITIAL. Der Prozess wurde noch nicht ausgeführt.

Optionen

ACQPROCESS

Gibt an, dass der Prozess, der zurzeit durch den Requester angefordert ist, zurückgesetzt werden soll.

Bedingungen

16 INVREQ

RESP2-Werte:

15

Die Arbeitseinheit, die die Anforderung abgesetzt hat, hat keinen Prozess angefordert.

17 IOERR

RESP2-Werte:

29

Die Repository-Datei ist nicht verfügbar.

30

Es ist ein Ein-/Ausgabefehler für die Repository-Datei aufgetreten.

100 LOCKED

Die Anforderung kann nicht ausgeführt werden, weil eine Sperre für den relevanten Datensatz in der Repository-Datei besteht.

70 NOTAUTH

RESP2-Werte:

101

Der Benutzer, der der absetzenden Task zugeordnet ist, ist nicht berechtigt, den Prozess zurückzusetzen.

106 PROCESSBUSY

RESP2-Werte:

13

Die Anforderung hat das Zeitlimit überschritten. Es kann sein, dass die Beendigung einer anderen Task, die diesen Prozessdatensatz verwendet, verhindert wurde.

108 PROCESSERR

RESP2-Werte:

14

Der zurückzusetzende Prozess befindet sich nicht im Modus COMPLETE oder INITIAL.

RESET ACTIVITY

Setzt eine BTS-Aktivität auf ihren Anfangsstatus zurück.

RESET ACTIVITY

➡ RESET — ACTIVITY(*datenwert*) ➡

Bedingungen: ACTIVITYBUSY, ACTIVITYERR, INVREQ, IOERR, LOCKED, NOTAUTH

Beschreibung

Der Befehl RESET ACTIVITY setzt eine untergeordnete BTS-Aktivität auf ihren Anfangsstatus zurück. Das zugehörige Beendigungsereignis wird dem Ereignispool der übergeordneten Aktivität hinzugefügt, wobei der Ausführungsstatus (FIRED) auf NOTFIRED (nicht ausgelöst) gesetzt wird. Wenn die Aktivität selbst untergeordnete Aktivitäten hat, werden diese gelöscht.

Anmerkung: Das Zurücksetzen hat keine Auswirkung auf die Datencontainer der Aktivität, die unverändert bleiben.

Setzen Sie diesen Befehl vor einem zweiten Befehl RUN ab, wenn eine Aktivität erneut versucht werden muss. Wenn die Aktivität erneut ausgeführt wird, wird ihr ein Ereignis DFHINITIAL gesendet.

Für eine Aktivität muss Folgendes zutreffen, damit sie zurückgesetzt werden kann:

1. Sie muss eine untergeordnete Aktivität der Aktivität sein, die den Befehl RESET absetzt.
2. Sie muss einen der folgenden Modi haben:
 - COMPLETE. Dies ist der Normalfall. Möglicherweise wurde die Aktivität abnormal beendet und muss zurückgesetzt werden, bevor sie erneut versucht wird.
 - INITIAL. Die Aktivität wurde noch nicht ausgeführt.

Optionen

ACTIVITY(datenwert)

Gibt den Namen (1 - 16 Zeichen) der Aktivität an, die zurückgesetzt werden soll. Dies muss eine untergeordnete Aktivität der aktuellen Aktivität sein.

Bedingungen

107 ACTIVITYBUSY

RESP2-Werte:

19

Die Anforderung hat das Zeitlimit überschritten. Es kann sein, dass die Beendigung einer anderen Task, die diesen Aktivitätsdatensatz verwendet, verhindert wurde.

109 ACTIVITYERR

RESP2-Werte:

8

Die in der Option ACTIVITY angegebene Aktivität ist keine untergeordnete Aktivität der aktuellen Aktivität.

14

Die zurückzusetzende Aktivität befindet sich nicht im Modus COMPLETE oder INITIAL.

16 INVREQ

RESP2-Werte:

4

Der Befehl RESET ACTIVITY wurde außerhalb des Gültigkeitsbereichs einer momentan aktiven Aktivität abgesetzt.

17 IOERR

RESP2-Werte:

29

Die Repository-Datei ist nicht verfügbar.

30

Es ist ein Ein-/Ausgabefehler für die Repository-Datei aufgetreten.

100 LOCKED

Die Anforderung kann nicht ausgeführt werden, weil eine Sperre für den relevanten Datensatz in der Repository-Datei besteht.

70 NOTAUTH

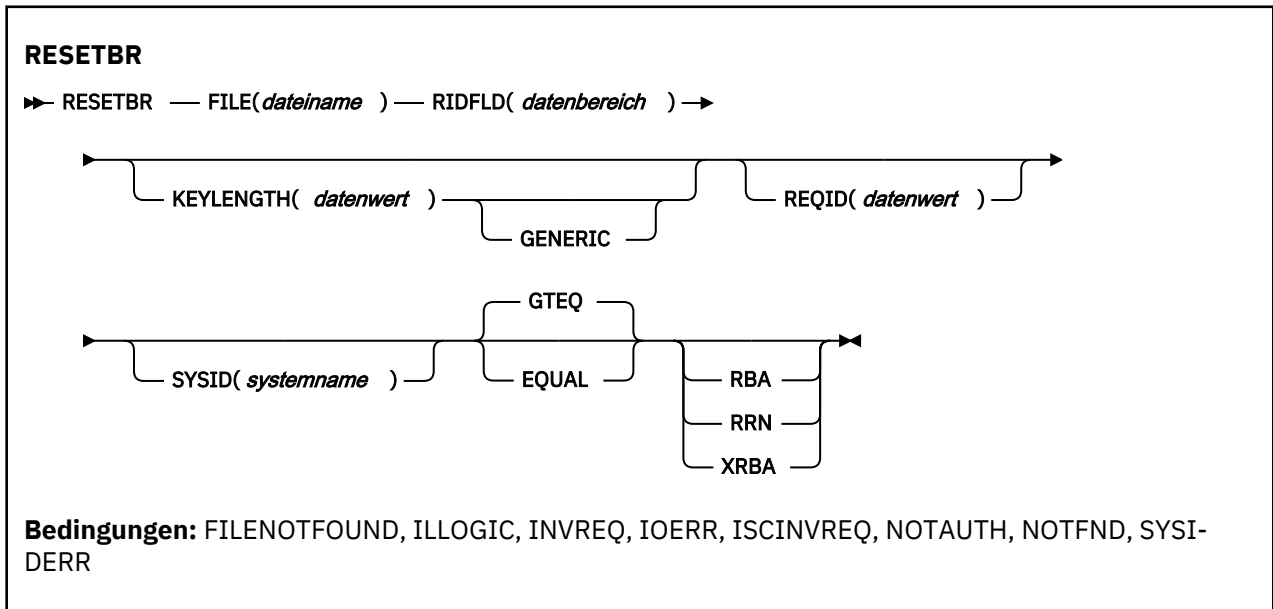
RESP2-Werte:

101

Der Benutzer, der der absetzenden Task zugeordnet ist, ist nicht berechtigt, die Aktivität zurückzusetzen.

RESETBR

Setzt den Start eines Suchvorgangs zurück.



Dieser Befehl ist threadsicher, wenn für die Datei, auf die er sich bezieht, Folgendes zutrifft:

- Sie ist als fern definiert und der Befehl wird über eine IPIC-Verbindung durch Funktionsverlagerung an eine ferne CICS-Region übertragen.
- Sie ist als lokale VSAM-Datei, als VSAM-RLS-Datei oder als Coupling-Facility-Datentabelle definiert.

Dieser Befehl ist nicht threadsicher, wenn für die Datei, auf die er sich bezieht, Folgendes zutrifft:

- Sie ist als fern definiert und der Befehl wird über eine Nicht-IPIC-Verbindung durch Funktionsverlagerung übertragen.
- Sie ist als gemeinsam genutzte Datentabelle oder als BDAM-Datei definiert.

Beschreibung

Der Befehl RESETBR gibt während eines Suchvorgangs den Datensatz in einer Datei oder Datentabelle auf einem lokalen oder fernen System an, bei dem der Suchvorgang neu positioniert werden soll.

Beim Durchsuchen einer VSAM-Datei oder -Datentabelle können Sie den Suchvorgang mit dem Befehl RESETBR neu positionieren (was auch durch eine Änderung des Datenbereichs für die Option RIDFLD bei einem Befehl READNEXT oder READPREV erreicht werden kann); darüber hinaus können Sie die Merkmale des Suchvorgangs ändern, die beim Befehl STARTBR angegeben wurden, ohne den Suchvorgang zu beenden. Diejenigen Merkmale, die geändert werden können, sind die durch die Optionen GENERIC, GTEQ und RBA angegebenen Merkmale.

Beim Durchsuchen einer BDAM-Datei können Sie diesen Befehl jederzeit vor Ausgabe eines weiteren Befehls für einen Suchvorgang einbeziehen. Er ähnelt einer Folge von Befehlen ENDBR und STARTBR (jedoch mit geringerer Funktionalität) und gestattet BDAM-Benutzer mit einer Funktionalität zum Überspringen von Sequenzen aus, die für VSAM-Benutzer bei Verwendung des Befehls READNEXT verfügbar ist.

Falls eine Anforderung RESETBR den exakten Schlüssel angibt, bei dem der Suchvorgang gestartet werden soll (sie also den vollständigen Schlüssel und das Schlüsselwort EQUAL angibt), wird beim folgenden

Befehl READNEXT (bzw. READPREV) möglicherweise nicht derselbe Datensatz zurückgegeben, der durch den Befehl RESETBR für eine im NSR- oder RLS-Modus von VSAM geöffnete Datei angegeben wird. Dies kann auftreten, weil der ursprünglich im Befehl RESETBR angegebene Datensatz zwischen der Fertigstellung des Befehls RESETBR und der Ausgabe eines Befehls READNEXT oder READPREV durch eine andere Transaktion gelöscht werden kann. Im LSR-Modus von VSAM kann der ursprüngliche Datensatz zwischen Befehlen RESETBR und READNEXT nicht gelöscht werden.

Anmerkung: Der Befehl RESETBR macht einen Wert für TOKEN, der durch einen vorherigen Befehl READ oder READNEXT festgelegt wurde, ungültig.

Optionen

EQUAL

Gibt an, dass die Suche nur durch einen Datensatz erfüllt wird, dessen (vollständiger oder generischer) Schlüssel ebenfalls in der Option RIDFLD angegeben ist.

FILE(*dateiname*)

(VSAM und Datentabelle) Gibt den Namen der Datei an, auf die zugegriffen werden soll.

Bei Angabe eines Wertes für SYSID wird davon ausgegangen, dass sich die Datei, auf die sich dieser Dateiname bezieht, auf einem fernen System befindet, wobei es ohne Belang ist, ob der Name für CICS definiert ist. Andernfalls wird mithilfe der Ressourcendefinition ermittelt, ob sich die Datei auf einem lokalen oder fernen System befindet.

GENERIC

(Schlüsseladressierte VSAM-Datei (KSDS), Pfad oder Datentabelle) Gibt an, dass es sich bei dem Suchkriterium um einen generischen Schlüssel handelt, dessen Länge in der Option KEYLENGTH angegeben ist. Die Suche nach einem Datensatz wird erfüllt, wenn ein Datensatz gefunden wird, der dieselben Anfangszeichen (generischer Schlüssel) wie hier angegeben hat.

GTEQ

(VSAM und Datentabelle) Gibt an, dass bei einer erfolglosen Suche nach einem Datensatz, der denselben (vollständigen oder generischen) Schlüssel wie in der Option RIDFLD angegeben aufweist, der erste Datensatz abgerufen wird, der einen größeren Schlüssel besitzt. Verwenden Sie diese Option nur bei Angabe eines Schlüssels oder einer relativen Satznummer.

KEYLENGTH(*datenwert*)

Gibt die Länge (Halbwort-Binärzahl) des Schlüssels an, der in der Option RIDFLD angegeben ist. Ausgenommen hiervon ist die Angabe der relativen Byteadresse (RBA) oder der relativen Satznummer (RRN); in diesem Fall ist die Option KEYLENGTH nicht gültig.

Diese Option muss bei Angabe von GENERIC angegeben werden; sie kann immer dann angegeben werden, wenn ein Schlüssel angegeben ist. Falls die angegebene Länge von der für die Datei definierten Länge abweicht und es sich nicht um eine generische Operation handelt, tritt die Bedingung INVREQ auf.

Die Bedingung INVREQ tritt ebenfalls auf, falls ein Befehl RESETBR die Option GENERIC angibt und der Wert von KEYLENGTH nicht kleiner als der Wert ist, der in der VSAM-Definition angegeben ist.

Falls die Option KEYLENGTH(0) verwendet wird, um den ersten Datensatz in der Datei zu lesen, muss auch die Option GTEQ angegeben werden. Falls EQUAL entweder explizit oder standardmäßig mit KEYLENGTH(0) angegeben wird, sind die Ergebnisse des Befehls STARTBR unvorhersehbar.

Für ferne Dateien kann der Wert von KEYLENGTH in der FILE-Definition angegeben werden. Falls die Option KEYLENGTH dort nicht definiert ist, nicht im Anwendungsprogramm angegeben ist und der Schlüssel länger als 4 Zeichen ist, lautet der Standardwert 4.

RBA

(Nur schlüsseladressierte VSAM-Dateien oder ESDS-basierte Dateien bzw. CICS-verwaltete Datentabellen, jedoch keine Pfade) Gibt an, dass das Feld für die Satznummer in der Option RIDFLD eine relative Byteadresse enthält. Verwenden Sie diese Option nur dann, wenn Sie eine ESDS- oder KSDS-Basis durchsuchen und die Datensätze nicht mit Schlüsseln, sondern mit relativen Byteadressen angeben.

Die Option RBA kann für Folgendes nicht verwendet werden:

- Benutzerverwaltete Datentabellen
- Coupling-Facility-Datentabellen
- Jede KSDS-Datei, die im RLS-Zugriffsmodus geöffnet ist
- KSDS-Dateien, die erweiterte Adressierung verwenden

Von einer Verwendung der Option RBA wird darüber hinaus bei ESDS-Dateien mit einer Größe von mehr als 4 GB abgeraten. (Verwenden Sie stattdessen XRBA.)

REQID(datenwert)

Gibt in Form eines Halbwort-Binärwertes eine eindeutige Anforderungskennung für einen Suchvorgang an, mit dessen Hilfe Mehrfachsuchvorgänge für eine Datei gesteuert werden. Falls diese Option nicht angegeben ist, wird vom Standardwert Null ausgegangen.

RIDFLD(datenbereich)

Gibt das Feld für die Satzkennung an. Der Inhalt kann ein Schlüssel, eine relative Byteadresse, eine relative Satznummer (für VSAM-Dateien) oder ein Blockverweis, ein physischer Schlüssel und ein Entblockungsargument (für BDAM-Dateien) sein. Bei einer relativen Byteadresse oder einer relativen Satznummer muss dieses Feld das Format einer Vollwort-Binärzahl besitzen. Für eine relative Byteadresse kann der Wert von RIDFLD größer-gleich 0 sein. Für eine relative Satznummer kann der Wert von RIDFLD größer-gleich 1 sein.

Bei VSAM gibt eine vollständige Datensatz-ID aus Werten X'FF' an, dass der Suchvorgang zur Vorbereitung einer Rückwärtssuche mit Befehlen READPREV am Ende der Datei positioniert werden soll.

RRN

(Schlüsseladressierte VSAM-Dateien) Gibt an, dass das in der Option RIDFLD angegebene Feld für die Satzkennung eine relative Satznummer enthält.

SYSID(systemname)

Gibt den Namen des Systems an, an das die Anforderung gerichtet ist.

Falls Sie die Option SYSID angeben und beide Optionen RBA und RRN weglassen, müssen Sie ebenfalls die Option KEYLENGTH angeben, weil diese nicht in der Ressourcendefinition zu finden ist.

XRBA

Gibt an, dass das in der Option RIDFLD angegebene Feld für die Satzkennung eine erweiterte relative Byteadresse enthält. Diese Option sollte bei Suchvorgängen für Datensätze in einer ESDS-Datei mit erweiterter Adressierung verwendet werden.

Sie können die Option XRBA bei einem Befehl RESETBR nur dann angeben, wenn im zugehörigen Befehl STARTBR ebenfalls XRBA angegeben war.

Bedingungen

12 FILENOTFOUND

RESP2-Werte:

1

Ein in der Option FILE referenzierter Dateiname ist nicht für CICS definiert.

Standardaktion: Die Task wird abnormal beendet.

21 ILLOGIC

RESP2-Werte (VSAM):

110

Es ist ein VSAM-Fehler aufgetreten, der zu keiner der anderen CICS-Antwortkategorien gehört.

Weitere Informationen sind bei EIBRCODE im EXEC-Schnittstellenblock verfügbar; Details finden Sie unter [EIB-Felder](#).

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

RESP2-Werte:

25

Die Optionen KEYLENGTH und GENERIC werden angegeben und die in der Option KEYLENGTH angegebene Länge ist größer oder gleich der Länge eines vollständigen Schlüssels.

26

Die Option KEYLENGTH wurde angegeben (jedoch nicht die Option GENERIC) und die angegebene Schlüssellänge ist nicht mit der Länge identisch, die für die Datei definiert ist, auf die sich diese Dateiangabe bezieht.

36

Der Wert für REQID bzw. SYSID oder der Dateiname stimmt nicht mit dem eines erfolgreichen Befehls STARTBR überein.

37

Der Typ der Satzkennung (z. B. Schlüssel oder relative Byteadresse), der für den Zugriff auf eine Datei während des Suchvorgangs verwendet wird, wurde geändert. Die Angabe eines Adressierungstyps bei einem Befehl STARTBR und die eines anderen Typs bei einem Befehl RESETBR ist nicht möglich.

42

Die Optionen KEYLENGTH und GENERIC werden angegeben und die in der Option KEYLENGTH angegebene Länge ist kleiner als null.

44

Der Befehl hat nicht das Format eines Befehls RESETBR für eine benutzerverwaltete oder Coupling-Facility-Datentabelle; beispielsweise ist die relative Byteadresse (RBA) angegeben.

51

Ein Befehl RESETBR für eine schlüsseladressierte Datei, auf die im RLS-Modus zugegriffen wird, gibt das Schlüsselwort RBA an. Der RLS-Modus unterstützt für Datensätze in schlüsseladressierten Dateien keinen Zugriff über die relative Byteadresse (RBA).

Standardaktion: Die Task wird abnormal beendet.

17 IOERR

RESP2-Werte:

120

Während der Dateisteuerungsoperation ist ein E/A-Fehler aufgetreten. Ein E/A-Fehler ist ein nicht erwartetes Ereignis, das nicht durch eine CICS-Bedingung abgedeckt ist.

Bei VSAM-Dateien gibt die Bedingung IOERR für gewöhnlich einen Hardwarefehler an. Weitere Informationen sind im EXEC-Schnittstellenblock verfügbar; Details finden Sie unter [EIB-Felder](#).

Bei einer Coupling-Facility-Datentabelle gibt eine Bedingung IOERR an, dass eine falsche Antwort von einem Coupling-Facility-Zugriff zurückgegeben wurde.

Standardaktion: Die Task wird abnormal beendet.

54 ISINVREQ

RESP2-Werte:

70

Das ferne System gibt einen Fehler an, der keiner bekannten Bedingung entspricht.

Standardaktion: Die Task wird abnormal beendet.

70 NOTAUTH

RESP2-Werte:

101

Eine Ressourcensicherheitsprüfung ist für FILE(*dateiname*) fehlgeschlagen.

Standardaktion: Die Task wird abnormal beendet.

13 NOTFND

RESP2-Werte:

80

Der Versuch, einen Datensatz anhand des angegebenen Sucharguments abzurufen, war erfolglos.

Die Bedingung NOTFND kann außerdem auftreten, wenn ein generischer Befehl RESETBR mit der Option KEYLENGTH(0) auch die Option EQUAL angibt.

81

Die Option XRBA war angegeben und der Wert von RIDFLD war größer als 4 GB, bei der Datei handelt es sich jedoch nicht um eine ESDS-Datei mit erweiterter Adressierung.

Standardaktion: Die Task wird abnormal beendet.

53 SYSIDERR

RESP2-Werte:

130

Die Option SYSID gibt einen Namen an, bei dem es sich weder um die lokale CICS-Region noch um ein fernes System handelt, das durch eine CONNECTION-Definition definiert ist. Die Bedingung SYSIDERR tritt außerdem auf, wenn die Verbindung zu dem fernen System geschlossen ist.

131

Für eine Coupling-Facility-Datentabelle ist die Verbindung zum Server für Coupling-Facility-Datentabellen fehlgeschlagen. Dies könnte dadurch verursacht worden sein, dass der Server selbst fehlgeschlagen ist, oder dass der Server zwar verfügbar ist, CICS jedoch keine Verbindung zu ihm herstellen konnte.

132

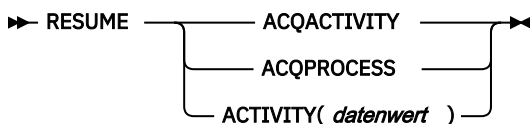
Der Befehl RESETBR wurde für eine Coupling-Facility-Datentabelle ausgegeben, die nicht mehr vorhanden ist. Vermutlich liegt dies an einem Fehler der Coupling-Facility, durch den der Server für Coupling-Facility-Datentabellen ebenfalls fehlschlägt. Informationen zum erneuten Starten eines Servers für Coupling-Facility-Datentabellen und zum erneuten Laden einer Tabelle finden Sie unter [Server für Coupling-Facility-Datentabellen einrichten und ausführen](#).

Standardaktion: Die Task wird abnormal beendet.

RESUME

Setzt einen ausgesetzten BTS-Prozess oder eine ausgesetzte BTS-Aktivität fort.

RESUME



Bedingungen: ACTIVITYBUSY, ACTIVITYERR, INVREQ, IOERR, LOCKED, PROCESSERR

Beschreibung

Der Befehl RESUME setzt einen BTS-Prozess oder eine BTS-Aktivität fort, der bzw. die zuvor (durch einen Befehl SUSPEND) ausgesetzt wurde. Das heißt, mit dem Befehl kann der Prozess oder die Aktivität erneut aktiviert werden, wenn Ereignisse im zugehörigen Ereignispool ausgelöst werden. Wenn Ereignisse, die normalerweise eine erneute Aktivierung (erneuten Anschluss) verursacht hätten, während der Dauer, die der Prozess oder die Aktivität ausgesetzt war, aufgetreten sind, wird die letztere für alle diese Ereignisse erneut angeschlossen (reaktiviert).

Der einzige Prozess, den ein Programm fortsetzen kann, ist derjenige, den es in der aktuellen Arbeitseinheit angefordert hat.

Die einzigen Aktivitäten, die ein Programm fortsetzen kann, sind die folgenden:

- Seine eigenen untergeordneten Aktivitäten, falls es als Aktivierung einer Aktivität ausgeführt wird. Es kann mehrere der eigenen untergeordneten Aktivitäten innerhalb derselben Arbeitseinheit fortsetzen.
- Die Aktivität, die es durch einen Befehl ACQUIRE ACTIVITYID in der aktuellen Arbeitseinheit angefordert hat.

Optionen

ACQACTIVITY

Gibt an, dass die fortzusetzende Aktivität diejenige ist, die von der aktuellen Arbeitseinheit durch einen Befehl ACQUIRE ACTIVITYID angefordert wurde.

ACQPROCESS

Gibt an, dass der Prozess, der zurzeit durch den Requester angefordert ist, fortgesetzt werden soll.

ACTIVITY(datenwert)

Gibt den Namen (1 - 16 Zeichen) der untergeordneten Aktivität an, die fortgesetzt werden soll.

Bedingungen

107 ACTIVITYBUSY

RESP2-Werte:

19

Die Anforderung hat das Zeitlimit überschritten. Es kann sein, dass die Beendigung einer anderen Task, die diesen Aktivitätsdatensatz verwendet, verhindert wurde.

109 ACTIVITYERR

RESP2-Werte:

8

Die in der Option ACTIVITY angegebene Aktivität konnte nicht gefunden werden.

14

Die Aktivität ist im Modus COMPLETE oder CANCELLING und kann daher nicht fortgesetzt werden.

16 INVREQ

RESP2-Werte:

4

Mit der Option ACTIVITY wurde eine untergeordnete Aktivität angegeben, aber der Befehl wurde außerhalb des Geltungsbereichs einer gegenwärtig aktiven Aktivität ausgegeben.

15

Die Option ACQPROCESS wurde verwendet, jedoch hat die Arbeitseinheit, die die Anforderung abgesetzt hat, keinen Prozess angefordert.

24

Die Option ACQACTIVITY wurde verwendet, jedoch hat die Arbeitseinheit, die die Anforderung abgesetzt hat, keine Aktivität angefordert.

17 IOERR

RESP2-Werte:

29

Die Repository-Datei ist nicht verfügbar.

30

Es ist ein Ein-/Ausgabefehler für die Repository-Datei aufgetreten.

100 LOCKED

Die Anforderung kann nicht ausgeführt werden, weil eine Sperre für den relevanten Datensatz in der Repository-Datei besteht.

108 PROCESSERR

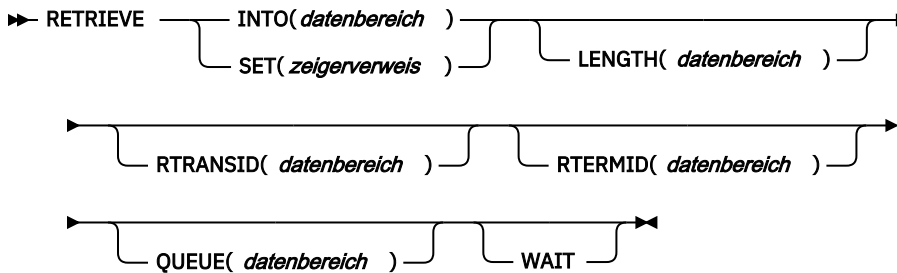
RESP2-Werte:

Der Prozess ist im Modus COMPLETE oder CANCELLING und kann daher nicht fortgesetzt werden.

RETRIEVE

Ruft Daten ab, die für eine Task gespeichert wurden.

RETRIEVE



Bedingungen: ENDDATA, ENVDEFERR, INVREQ, IOERR, LENGERR

Hinweis für dynamisches Transaktionsrouting: Die Verwendung des Befehls RETRIEVE mit der Option WAIT kann transaktionsübergreifende Affinitäten erzeugen, die die Nutzung des dynamischen Transaktionsroutings beeinträchtigen. Weitere Informationen zu Transaktionsaffinitäten finden Sie unter [Affinität](#).

Beschreibung

Der Befehl RETRIEVE ruft Daten ab, die durch abgelaufene Befehle START gespeichert wurden. Er stellt das einzige verfügbare Verfahren für den Zugriff auf solche Daten dar.

Eine Task, die keinem Terminal zugeordnet ist, kann lediglich auf den einen Datensatz zugreifen, der dem ursprünglichen Befehl START zugeordnet ist; zu diesem Zweck wird ein Befehl RETRIEVE ausgegeben. Der belegte Speicher für die Daten, die der Task zugeordnet sind, wird normalerweise bei der Ausführung des Befehls RETRIEVE oder bei der Beendigung der Task freigegeben, sofern vor der Beendigung kein Befehl RETRIEVE ausgeführt wird.

Falls im Befehl START die Option ATTACH angegeben wurde, wird der Speicher nicht freigegeben. (Die Option ASSIGN STARTCODE in einer solchen Task gibt 'U' anstelle von 'S' oder 'SD' zurück).

Eine Task, die einem Terminal zugeordnet ist, kann auf alle Datensätze zugreifen, die allen abgelaufenen Befehlen START zugeordnet sind und dieselbe Transaktions-ID und Terminal-ID wie die Task besitzen, die den Befehl RETRIEVE ausgibt; hierzu werden aufeinanderfolgende Befehle RETRIEVE ausgegeben. Abgelaufene Datensätze werden für die Task auf Anforderung in der Reihenfolge ihrer Ablaufzeit ausgegeben, beginnend mit allen Daten, die durch den Befehl gespeichert wurden, von dem die Task gestartet wurde, und einschließlich der Daten aus allen Befehlen, die seit dem Start der Task abgelaufen sind. Jeder Datensatz wird aus dem temporären Speicher abgerufen, wobei der Wert von REQID für den ursprünglichen Befehl START als Kennung des Datensatzes im temporären Speicher verwendet wird.

Nachdem alle Datensätze abgerufen wurden, tritt die Bedingung ENDDATA auf. Der Speicher, mit dem der eine Datensatz belegt ist, der einem Befehl START zugeordnet ist, wird freigegeben, nachdem die Daten durch einen Befehl RETRIEVE abgerufen wurden; Speicher, der durch nicht abgerufene Daten belegt ist, wird freigegeben, wenn das CICS-System beendet wird.

Falls die abgerufenen Daten Funktionsverwaltungsholder (Function Management Header, FMH) enthalten, was durch die Option FMH im zugehörigen Befehl START angegeben ist, wird das Feld EIBFMH im EIB auf X'FF' gesetzt. Wenn kein Funktionsverwaltungsholder vorhanden ist, wird das Feld EIBFMH auf X'00' gesetzt.

Optionen

INTO(datenbereich)

Gibt den Benutzerdatenbereich an, in den die abgerufenen Daten geschrieben werden sollen.

LENGTH(datenbereich)

Gibt einen Halbwort-Binärwert an, um die Länge des Datenbereichs zu definieren, in den die abgerufenen Daten geschrieben werden.

Falls Sie die Option INTO angeben, muss das Argument ein Datenbereich sein, der die maximale Länge von Daten angibt, auf deren Verarbeitung das Programm vorbereitet ist. Falls der angegebene Wert kleiner als Null ist, wird Null angenommen. Falls die Länge der Daten den angegebenen Wert überschreitet, werden die Daten auf diesen Wert abgeschnitten und die Bedingung LENGERR tritt auf. Nach Abschluss der Abrufoperation wird der Datenbereich auf die ursprüngliche Länge der Daten gesetzt.

Bei Angabe der Option SET muss das Argument ein Datenbereich sein. Nach Abschluss der Abrufoperation wird der Datenbereich auf die Länge der Daten gesetzt.

Eine Beschreibung einer sicheren Obergrenze finden Sie unter „[LENGTH-Optionen in CICS-Befehlen](#)“ auf Seite 10.

QUEUE(datenbereich)

Gibt den acht Zeichen umfassenden Bereich für den Namen der Warteschlange für temporären Speicher an, auf den die Transaktion durch eine Ausgabe des Befehls RETRIEVE zugreifen kann.

RTERMID(datenbereich)

Gibt einen vier Zeichen umfassenden Bereich an, der in der Option TERMID eines möglicherweise nachfolgend ausgeführten Befehls START verwendet werden kann.

RTRANSID(datenbereich)

Gibt einen vier Zeichen umfassenden Bereich an, der in der Option TRANSID eines möglicherweise nachfolgend ausgeführten Befehls START verwendet werden kann.

SET(zeigerverweis)

Gibt den Zeigerverweis an, der auf die Adresse der abgerufenen Daten gesetzt werden soll.

Falls dem Anwendungsprogramm DATALOCATION(ANY) zugeordnet ist, kann die Adresse der Daten ober- oder unterhalb der 16-MB-Grenze liegen.

Wenn dem Anwendungsprogramm DATALOCATION(BELOW) zugeordnet ist und sich die Daten oberhalb der 16-MB-Grenze befinden, werden die Daten an eine Adresse unterhalb der 16-MB-Grenze kopiert und die Adresse dieser Kopie wird zurückgegeben.

Wenn für die aktive Task TASKDATAKEY(USER) angegeben wird und der Speicherschutz aktiv ist, befinden sich die zurückgegebenen Daten in einem Benutzerschlüssel. Wenn für die aktive Task TASKDATAKEY(CICS) angegeben wird und der Speicherschutz aktiv ist, befinden sich die zurückgegebenen Daten in einem CICS-Schlüssel.

Bei Verwendung von SET müssen Sie ebenfalls die Option LENGTH einbeziehen.

WAIT

Gibt an, dass die Task in einen Wartestatus versetzt wird, bis weitere abgelaufene Datensätze verfügbar werden, falls alle abgelaufenen Datensätze bereits abgerufen worden sind. Auch wenn dies bedeutet, dass die Bedingung ENDDATA nicht in dem Moment auftritt, in dem der Befehl RETRIEVE ausgegeben wird, tritt die Bedingung jedoch später auf, falls CICS beendet wird oder für die Task ein Deadlock-Zeitlimit auftritt und der Wartestatus der Task länger als das Deadlock-Zeitlimitintervall ist. (Weitere Informationen hierzu finden Sie im Abschnitt über die Option DTIMOUT von RDO DEFINE TRANSACTION.)

Der Versuch, einen Befehl RETRIEVE mit der Option WAIT während des Systemabschlusses auszugeben, führt zu einem Abbruch des Typs AICB, falls zur Erfüllung der Anforderung noch kein Datensatz verfügbar ist.

Bei Verwendung von WAIT müssen Sie mindestens eine weitere Option angeben.

Bedingungen**29 ENDDATA**

Tritt in einer der folgenden Situationen auf:

- Für die Task, die einen Befehl RETRIEVE ausgibt, sind keine weiteren Daten gespeichert. In diesem Fall kann die Bedingung als normale Antwort für das Dateiende beim sequenziellen Abrufen von Datensätzen betrachtet werden.
- Der Befehl RETRIEVE wird durch eine Task ausgegeben, die durch einen Befehl START gestartet wurde, der keine der Datenoptionen FROM, RTRANSID, RTERMID oder QUEUE angegeben hat.
- Der Befehl RETRIEVE wird durch eine Nichtterminaltask ausgegeben, die nicht infolge eines Befehls START erstellt wurde.
- Die Option WAIT war angegeben und die Task wartete auf einen Datensatz, aber vor dem Deadlock-Zeitlimitintervall wurde kein Datensatz verfügbar (weitere Informationen enthält der Abschnitt über die Option DTIMOUT von RDO DEFINE TRANSACTION).
- Die Option WAIT war angegeben und die Task befand sich im Wartezustand, als der Systemabschluss für CICS eingeleitet wurde. Der Versuch, einen Befehl RETRIEVE mit der Option WAIT während des Systemabschlusses auszugeben, führt zu einem Abbruch des Typs AICB, falls zur Erfüllung der Anforderung noch kein Datensatz verfügbar ist.
- Ein Befehl RETRIEVE mit der Option WAIT wurde ausgegeben, als keine Daten verfügbar waren; die Task wurde durch einen Befehl START initialisiert, der in der Option TERMID eine APPC-Verbindung oder ein Terminal angegeben hat.

Standardaktion: Die Task wird abnormal beendet.

56 ENVDEFERR

Tritt auf, wenn ein Befehl RETRIEVE eine Option angibt, die im korrespondierenden Befehl START nicht angegeben war.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

Tritt auf, falls der Befehl RETRIEVE nicht für die Verarbeitung durch CICS gültig ist.

Standardaktion: Die Task wird abnormal beendet.

17 IOERR

Tritt auf, falls während einer Operation RETRIEVE ein Ein-/Ausgabefehler vorliegt. Die Operation kann durch eine erneute Ausgabe des Befehls RETRIEVE wiederholt werden.

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

Tritt auf, falls die angegebene Länge kleiner als die tatsächliche Länge der gespeicherten Daten ist.

Standardaktion: Die Task wird abnormal beendet.

Beispiele

Das folgende Beispiel zeigt, wie durch einen Befehl START für die Task gespeicherte Daten abgerufen und im vom Benutzer angegebenen Datenbereich namens DATAFLD gespeichert werden.

```
EXEC CICS RETRIEVE
      INTO(DATAFLD)
      LENGTH(LENG)
```

Das folgende Beispiel zeigt, wie das Abrufen eines Datensatzes angefordert wird, der für eine Task in einem durch CICS bereitgestellten Datenbereich gespeichert wurde. Der durch die Option SET angegebene Zeigerverweis (PREF) wird auf die Adresse des Speicherbereichs gesetzt, der für den Datensatz reserviert ist.

```
EXEC CICS RETRIEVE  
      SET(PREF)  
      LENGTH(LENG)
```

RETRIEVE REATTACH EVENT

Ruft den Namen eines Ereignisses ab, das bewirkt hat, dass die aktuelle BTS-Aktivität wieder angeschlossen (reaktiviert) wurde.

RETRIEVE REATTACH EVENT

➔ RETRIEVE — REATTACH — EVENT(*datenbereich*) —
EVENTTYPE(*cvda*)

Bedingungen: END, INVREQ

Beschreibung

Der Befehl RETRIEVE REATTACH EVENT hat folgende Funktionsweise:

- Er gibt den Namen des nächsten Ereignisses in der Reaktivierungswarteschlange der aktuellen BTS-Aktivität zurück.
- Wenn das abgerufene Ereignis atormar ist, setzt der Befehl den Ausführungsstatus (FIRED) auf NOTFIRED zurück. (Zusammengesetzte Ereignisse werden durch diesen Befehl nicht zurückgesetzt, sondern es werden nur ihre Prädikate falsch.)

Verwenden Sie diesen Befehl, um den Namen des Ereignisses zu ermitteln, das bewirkt hat, dass die Aktivität erneut angeschlossen (reaktiviert) wurde. In einigen Fällen kann das erneute Anschließen daher rühren, dass mehr als ein Ereignis ausgelöst wurde, zum Beispiel wenn die Aktivität zuvor ausgesetzt wurde und Reaktivierungseignisse aufgetreten sind, während sie ausgesetzt war, oder wenn zwei oder mehr Zeitgeberereignisse gleichzeitig ausgelöst wurden. Der Ereignisname bzw. die Ereignisnamen werden in die Reaktivierungswarteschlange eingefügt, aus der sie durch einen oder mehrere Befehle RETRIEVE REATTACH abgerufen werden können.

Jedes Mal, wenn eine Aktivität aktiviert wird, muss sie mindestens ein Reaktivierungseignis verarbeiten. Das heißt, sie muss mindestens einen Befehl RETRIEVE REATTACH EVENT absetzen und (falls dies nicht automatisch durch CICS geschieht) den Ausführungsstatus des abgerufenen Ereignisses auf NOTFIRED zurücksetzen (siehe [Reaktivierungseignisse zurücksetzen und löschen](#)). Wenn dies nicht geschieht, führt dies dazu, dass die Aktivität abnormal beendet wird, weil sie keinen Fortschritt gemacht hat: sie hat keine Reaktivierungseignisse zurückgesetzt und läuft daher Gefahr, in eine unbeabsichtigte Schleife zu geraten.

Wenn sich mehrere Ereignisse in der Reaktivierungswarteschlange einer Aktivität befinden, kann die Aktivität durch das Absetzen mehrerer Befehle RETRIEVE REATTACH EVENT mehrere oder alle Ereignisse in einer einzigen Aktivierung verarbeiten. Alternativ kann sie sie einzeln verarbeiten, indem sie nur einen Befehl RETRIEVE pro Aktivierung absetzt und zurückkehrt. Anschließend wird sie erneut aktiviert, um das nächste Ereignis in ihrer Reaktivierungswarteschlange zu verarbeiten. Welchen Ansatz Sie wählen, hängt vom Programmentwurf ab. Beachten Sie, dass bei der Verarbeitung mehrerer Reaktivierungseignisse in derselben Aktivierung ein Synchronisationspunkt erst stattfindet, wenn die Aktivierung zurückkehrt.

Anmerkung: Durch das Abrufen eines zusammengesetzten Ereignisses aus der Reaktivierungswarteschlange wird der Status des zusammengesetzten Ereignisses nicht auf NOTFIRED zurückgesetzt. Wenn die Aktivität ein zusammengesetztes Reaktivierungseignis abrufen muss, muss das Aktivitätsprogramm deshalb möglicherweise einen oder mehrere Befehle RETRIEVE SUBEVENT absetzen, um das Unterereignis oder die Unterereignisse, die ausgelöst wurden, abzurufen (und zurückzusetzen). Dies hat wiederum zur Folge, dass der Ausführungsstatus des zusammengesetzten Ereignisses erneut ausgewertet wird.

Optionen

EVENT(datenbereich)

Gibt den 16 Zeichen langen Namen des Ereignisses zurück, das bewirkt hat, dass diese Aktivität erneut angeschlossen (reaktiviert) wird.

EVENTTYPE(cvda)

Gibt den Typ des Reaktivierungsereignisses zurück. Die CVDA-Werte lauten wie folgt:

ACTIVITY

Aktivitätsbeendigungsereignis

COMPOSITE

Zusammengesetztes Ereignis

INPUT

Eingabeereignis

SYSTEM

Das BTS-Systemereignis DFHINITIAL.

TIMER

Zeitgeberereignis

Bedingungen

83 END

RESP2-Werte:

8

Es sind keine weiteren Ereignisse zum Abrufen vorhanden.

16 INVREQ

RESP2-Werte:

1

Der Befehl wurde außerhalb des Gültigkeitsbereichs einer Aktivität abgesetzt.

RETRIEVE SUBEVENT

Ruft den Namen des nächsten Unterereignisses in der Unterereigniswarteschlange eines zusammengesetzten BTS-Ereignisses ab.

RETRIEVE SUBEVENT

➡ RETRIEVE — SUBEVENT(*datenbereich*) — EVENT(*datenwert*) — EVENTTYPE(*cvda*)

Bedingungen: END, EVENTERR, INVREQ

Beschreibung

Der Befehl RETRIEVE SUBEVENT hat folgende Funktionsweise:

- Er ruft den Namen des nächsten Unterereignisses in der Unterereigniswarteschlange eines zusammengesetzten BTS-Ereignisses ab.
- Er setzt den Ausführungsstatus (FIRED) des abgerufenen Unterereignisses auf NOTFIRED.
- Er bewirkt, dass der Ausführungsstatus des zusammengesetzten Ereignisses erneut ausgewertet wird.

Das Auslösen eines zusammengesetzten Ereignisses geschieht durch das Auslösen einer Gruppe aus einem oder mehreren Unterereignissen. Die Namen von Unterereignissen, die ausgelöst wurden, werden in die Unterereigniswarteschlange des zusammengesetzten Ereignisses eingereiht, aus der sie in der Reihenfolge durch aufeinanderfolgende Befehle RETRIEVE SUBEVENT abgerufen werden können.

Mithilfe dieses Befehls können Sie ermitteln, welches Unterereignis oder welche Unterereignisse dazu führten, dass ein zusammengesetztes Ereignis ausgelöst wurde.

Anmerkung:

1. Das Vorhandensein von Ereignissen in der Unterereigniswarteschlange impliziert nicht, dass das zusammengesetzte Ereignis ausgelöst wurde. (Einige Unterereignisse in der Gruppe, die erforderlich ist, um das zusammengesetzte Ereignis auszulösen, befinden sich möglicherweise immer noch im Status NOTFIRED (nicht ausgelöst) und befinden sich nicht in der Unterereigniswarteschlange.) Zum Ermitteln, ob ein zusammengesetztes Ereignis ausgelöst wurde, verwenden Sie den Befehl TEST EVENT.
2. Das Abrufen ist einmalig. Wenn der Name eines ausgelösten Unterereignisses abgerufen wurde, kann dieses Unterereignis nicht noch einmal abgerufen werden.
3. Da der Befehl RETRIEVE SUBEVENT den Ausführungsstatus des Unterereignisses zurücksetzt, hat er zur Folge, dass der Ausführungsstatus des zusammengesetzten Ereignisses erneut ausgewertet wird.

Optionen

EVENT(datenwert)

Gibt den Namen (1 - 16 Zeichen) des zusammengesetzten Ereignisses an.

EVENTTYPE(cvda)

Gibt den Typ des Unterereignisses zurück. Die CVDA-Werte lauten wie folgt:

ACTIVITY

Aktivitätsbeendigungsereignis

INPUT

Eingabeereignis

TIMER

Zeitgeberereignis

SUBEVENT(datenbereich)

Gibt den 16 Zeichen langen Namen des Unterereignisses am Kopf der Unterereigniswarteschlange zurück.

Bedingungen

83 END

RESP2-Werte:

9

Es sind keine weiteren Unterereignisse zum Abrufen vorhanden.

10

Das zusammengesetzte Ereignis enthält keine Unterereignisse (es ist leer).

111 EVENTERR

RESP2-Werte:

4

Das in der Option EVENT angegebene Ereignis wird von BTS nicht erkannt.

16 INVREQ

RESP2-Werte:

1

Der Befehl wurde außerhalb des Gültigkeitsbereichs einer Aktivität abgesetzt.

2

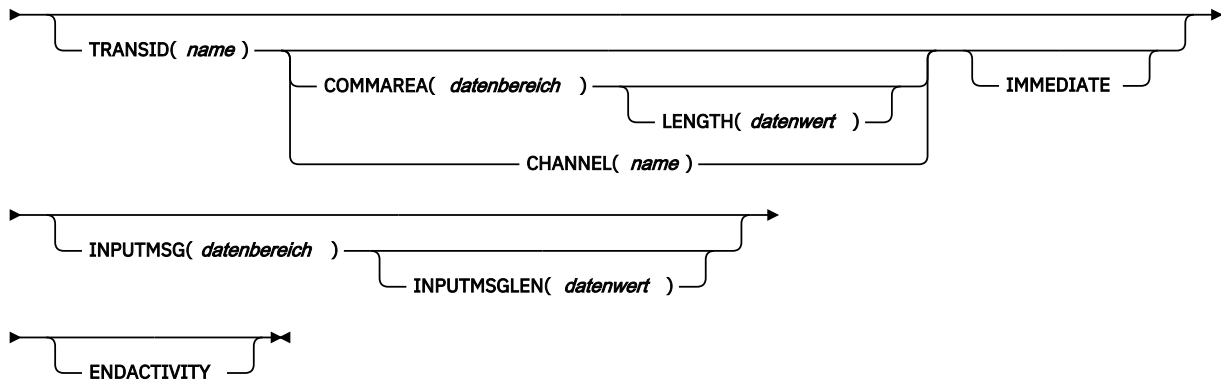
Das in der Option EVENT angegebene Ereignis ist ungültig. Es ist kein zusammengesetztes Ereignis.

RETURN

Gibt die Programmsteuerung zurück.

RETURN

→ RETURN →



Bedingungen: CHANNELERR, INVREQ, LENGERR

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl RETURN gibt die Steuerung von einem Anwendungsprogramm entweder an ein Anwendungsprogramm auf der nächsthöheren logischen Ebene oder an CICS zurück.

Bei der Rückgabe eines Kommunikationsbereichs (COMMAREA) gibt die Option LENGTH die Länge der zu übergebenden Daten an. Der übergebene Wert für LENGTH darf nicht größer als die Länge des Datenbereichs sein, der in der Option COMMAREA angegeben ist. Andernfalls sind die Ergebnisse unvorhersehbar und können zu einer Bedingung LENGERR führen, die im Abschnitt über die Übergabe von Daten an andere Programme unter [Daten an andere Programme übergeben](#) beschrieben ist.

Der gültige Bereich für die Länge von COMMAREA beträgt 0 bis 32763 Byte. Falls die angegebene Länge außerhalb dieses Bereichs liegt, tritt die Bedingung LENGERR auf.

Die Optionen COMMAREA, IMMEDIATE und CHANNEL können nur dann verwendet werden, wenn der Befehl RETURN die Steuerung an CICS zurückgibt; andernfalls tritt die Bedingung INVREQ auf.

Für den Befehl RETURN TRANSID findet keine Ressourcensicherheitsprüfung statt. Die Transaktionsicherheit ist jedoch trotzdem verfügbar, wenn CICS die zurückgegebene Transaktion zuordnet.

Optionen

CHANNEL(name)

Gibt den 1 bis 16 Zeichen langen Namen eines Kanals an, der für das nächste Programm verfügbar gemacht werden soll, das die Steuerung erhält. Zulässige Zeichen: A-Z a-z 0-9 \$ @ # / % & ? ! : | " = ~ , ; < > . - und _ . Führende und eingebettete Leerzeichen sind nicht zulässig. Falls der angegebene Name weniger als 16 Zeichen umfasst, wird er mit nachgestellten Leerzeichen auf 16 Zeichen aufgefüllt. Wenn der Kanal nicht vorhanden ist, wird er erstellt. Dieser neue Kanal bleibt gültig, bis sich die Verbindungsebene ändert. Informationen zum Geltungsbereich von Kanälen finden Sie unter [Geltungsbereich eines Kanals](#).

Kanalnamen werden immer in EBCDIC angegeben. Die oben angegebene Gruppe der zulässigen Zeichen für Kanalnamen enthält einige Zeichen, die nicht in allen EBCDIC-Codepages dieselbe Darstellung besitzen. Es wird daher empfohlen, bei der Benennung lediglich die Zeichen A-Z a-z 0-9 & : = , ; < > . - und _ zu verwenden, falls ein Kanal zwischen Regionen verlagert werden soll (es sich bei der in der Option TRANSID angegebenen Transaktion also um eine ferne Transaktion handelt).

Sie können den Kanalnamen DFHTRANSACTION angeben, um einen Transaktionskanal zu verwenden. Ein Transaktionskanal verliert nicht seine Gültigkeit, wenn sich die Verbindungsebene ändert; der Zu-

griff auf diesen Kanal ist in der Transaktion immer möglich. Weitere Informationen finden Sie unter Kanäle und Container.

Das Programm, das den Befehl RETURN ausgibt, kann Folgendes vornehmen:

- Es kann den Kanal zuvor durch einen oder mehrere Befehle PUT CONTAINER CHANNEL erstellen.
- Es kann seinen aktuellen Kanal durch dessen Namen angeben.
- Es kann einen nicht vorhandenen Kanal benennen, was zur Erstellung eines neuen und leeren Kanals führt.

Diese Option ist nur bei einem Befehl RETURN gültig, der durch ein Programm auf der höchsten logischen Ebene ausgegeben wird, also durch ein Programm, das die Steuerung an CICS zurückgibt.

COMMAREA(datenbereich)

Gibt einen Kommunikationsbereich an, der für das nächste Programm verfügbar gemacht werden soll, das die Steuerung erhält. In einem COBOL-Empfangsprogramm muss dieser Datenbereich den Namen DFHCOMMAREA erhalten. Weitere Informationen zum CICS-Kommunikationsbereich finden Sie unter Daten in Transaktionen gemeinsam nutzen. Da der Datenbereich vor dem Start des nächsten Programms freigegeben wird, wird eine Kopie des Datenbereichs erstellt und ein Zeiger auf die Kopie übergeben.

Der angegebene Kommunikationsbereich wird an das nächste Programm übergeben, das am Terminal ausgeführt wird. Beziehen Sie die Option IMMEDIATE ein, um sicherzustellen, dass der Kommunikationsbereich an das richtige Programm übergeben wird.

Diese Option ist nur bei einem Befehl RETURN gültig, der durch ein Programm auf der höchsten logischen Ebene ausgegeben wird, also durch ein Programm, das die Steuerung an CICS zurückgibt.

ENDACTIVITY

Diese Option ist für Programme bestimmt, die Aktivitäten von CICS Business Transaction Services (CICS BTS, CICS-Geschäftstransaktionsservices) implementieren. Sie gibt an, dass die aktuelle Aktivität gerade abgeschlossen wird und nicht reaktiviert werden soll.

Falls der Ereignispool der Aktivität keine Benutzerereignisse enthält, wird die Aktivität normal abgeschlossen.

Wenn der Ereignispool der Aktivität Benutzerereignisse enthält (ausgelöst oder nicht ausgelöst), findet Folgendes statt:

- Falls es sich bei einem oder mehreren Ereignissen um Aktivitätsbeendigungsereignisse handelt, wird die Aktivität abgebrochen. Der Versuch, den Abschluss einer Aktivität zu erzwingen, bevor sie eine oder mehrere ihrer untergeordneten Aktivitäten verarbeitet hat, ist ein Fehler in der Programmlogik.
- Falls es sich bei keinem der Ereignisse um Aktivitätsbeendigungsereignisse handelt, werden die Ereignisse gelöscht und die Aktivität wird normal abgeschlossen.

Informationen zu BTS im Allgemeinen und zur Option ENDACTIVITY im Besonderen finden Sie unter Beendigung von Aktivitäten.

Außerhalb der Umgebung von CICS BTS wird diese Option ignoriert.

IMMEDIATE

Stellt sicher, dass die in der Option TRANSID angegebene Transaktion ungeachtet aller möglicherweise durch ATI für dieses Terminal eingereihten anderen Transaktionen als nächste Transaktion zugeordnet wird. Die nächste Transaktion wird sofort gestartet; für den Bediener stellt sich das Ganze so dar, als ob sie durch Terminaldaten gestartet worden wäre. Falls das Terminal das Klammerprotokoll verwendet, wird auch das Terminal in Klammern angegeben. Diese Option ist nur bei einem Befehl RETURN gültig, der durch ein Programm auf der höchsten logischen Ebene ausgegeben wird, also durch ein Programm, das die Steuerung an CICS zurückgibt.

In einer Mehrregionenumgebung ist zu beachten, dass die Verwendung der Option IMMEDIATE keinen Einfluss auf die Transaktionsdefinition hat, da sich diese weiterhin in der Terminal-Ownning Region (TOR, Terminalverwaltungsregion) befindet.

INPUTMSG(datenbereich)

Gibt Daten an, die entweder an eine andere Transaktion (angegeben durch die Option TRANSID) oder an ein aufrufendes Programm in einer Transaktion mit mehreren Programmen übergeben werden sollen. Sie können die Option INPUTMSG auch bei Rückgabe der Steuerung aus einem benutzerdefinierten Programm für dynamisches Transaktionsrouting an CICS verwenden, wenn Sie die ursprüngliche Eingabe ändern wollen.

In allen Fällen werden die Daten in dem durch INPUTMSG bezeichneten Datenbereich an das erste Programm übergeben, das nach dem Befehl RETURN einen Befehl RECEIVE ausgibt.

Weitere Informationen und Darstellungen zur Verwendung der Option INPUTMSG finden Sie unter [INPUTMSG](#).

INPUTMSGLEN(datenwert)

Gibt einen Halbwort-Binärwert an, der für die Option INPUTMSG verwendet werden soll.

LENGTH(datenwert)

Gibt einen Halbwort-Binärwert an, der die Länge der Byte im Kommunikationsbereich (Option COMMAREA) ist. Eine Beschreibung einer sicheren Obergrenze finden Sie unter [LENGTH-Optionen in CICS-Befehlen](#).

TRANSID(name)

Gibt die 1 bis 4 Zeichen lange Transaktions-ID an, die für die nächste Eingabenachricht verwendet werden soll, die an demjenigen Terminal eingegeben wird, dem die Task zugeordnet ist, von der der Befehl RETURN ausgegeben wurde. Der angegebene Name muss für CICS als Transaktion definiert sein.

Falls die Option TRANSID für ein Programm angegeben wird, das auf einem Terminal mit einer definierten permanenten Transaktions-ID ausgeführt wird, wird als Nächstes die permanente Transaktion des Terminals und nicht die im Befehl RETURN angegebene Transaktion initialisiert.

Falls Sie für TRANSID einen aus binären Nullen bestehenden Wert angeben, kann die Transaktions-ID für das nächste Programm, das dem Terminal zugeordnet werden soll, aus nachfolgenden Eingaben am Terminal ermittelt werden. Die Ausgabe eines Befehls RETURN mit einem aus binären Nullen bestehenden Wert für TRANSID sowie einem Wert für COMMAREA kann unvorhersehbare Ergebnisse erzeugen, falls die nächste Transaktion nicht für die Verarbeitung der Option COMMAREA codiert ist oder einen nicht für sie bestimmten Kommunikationsbereich empfängt.

Falls Sie die Option TRANSID für ein Programm angeben, das kein Programm der höchsten Ebene ist, und anschließend ein Fehler für COMMAREA, INPUTMSG oder CHANNEL im letzten Befehl RETURN auftritt, wird die Transaktions-ID gelöscht.

Die nächste Transaktions-ID wird auch bei einer abnormalen Beendigung der Transaktion gelöscht.

Falls zusammen mit dieser Option die Option IMMEDIATE angegeben ist, wird die Steuerung an die Transaktion übergeben, die in der Option TRANSID angegeben ist; diese Transaktion erhält Vorrang vor allen durch ATI eingereihten Transaktionen.

Falls die Option IMMEDIATE nicht zusammen mit dieser Option angegeben ist, wird diese Option durch eine von ATI initialisierte Transaktion identischen Namens, die in das Terminal eingereiht wird, auf einen Leerwert gesetzt.

Diese Option ist nicht gültig, falls die Transaktion, die den Befehl RETURN ausgibt, keinem Terminal oder einer logischen APPC-Einheit zugeordnet ist.

Bedingungen**122 CHANNELERR**

RESP2-Werte:

1

Der in der Option CHANNEL angegebene Name enthält ein ungültiges Zeichen oder eine ungültige Zeichenkombination.

16 INVREQ

RESP2-Werte:

1

Ein Befehl RETURN mit der Option TRANSID wurde in einem Programm ausgegeben, das keinem Terminal zugeordnet ist.

2

Ein Befehl RETURN mit der Option CHANNEL, COMMAREA oder IMMEDIATE wurde durch ein Programm ausgegeben, das sich nicht auf der höchsten logischen Ebene befindet.

4

Ein Befehl RETURN mit der Option TRANSID wurde in einem Programm ausgegeben, das einer logischen APPC-Einheit zugeordnet ist.

8

Ein Befehl RETURN mit der Option INPUTMSG wurde für ein Programm ausgegeben, das keinem Terminal zugeordnet ist oder das einer logischen APPC-Einheit bzw. einer IRC-Sitzung zugeordnet ist.

30

Die PG-Domäne wurde nicht initialisiert. In PLT-Programmen der ersten Phase sind Parameter für die EXEC-Anweisung RETURN nicht zulässig.

200

Ein Befehl RETURN wurde mit einer Option INPUTMSG durch ein von DPL aufgerufenes Programm ausgegeben.

203

Die Option CHANNEL war angegeben, aber die ferne Region, an die die Steuerung zurückgegeben wird, unterstützt Kanäle nicht.

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

RESP2-Werte:

11

Die Länge des Wertes für COMMAREA ist kleiner als 0 oder größer als 32763.

26

Als Adresse für COMMAREA wurde Null übergeben, aber die Länge für COMMAREA war ungleich null.

27

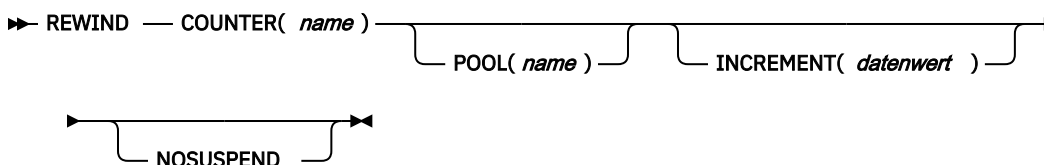
Die Länge des Wertes für INPUTMSG ist kleiner als 0 oder größer als 32767.

Standardaktion: Die Task wird abnormal beendet.

REWIND COUNTER und REWIND DCOUNTER

Spult einen benannten Zähler zurück, der seinen Grenzwert erreicht hat (dies bedeutet, dass die höchste Zahl zugewiesen wurde). COUNTER wird für Vollwortzähler mit Vorzeichen verwendet, DCOUNTER für Doppelwortzähler ohne Vorzeichen.

REWIND COUNTER



Bedingungen: BUSY, INVREQ, SUPPRESSED

Dieser Befehl ist threadsicher.

```

sequenceDiagram
    participant User
    participant DB as Database
    User->>DB: REWIND — DOUNTER( name )
    activate DB
    DB->>DB: POOL( name )
    DB->>DB: INCREMENT( datenbereich )
    deactivate DB
    User->>DB: NOSUSPEND
    activate DB
    deactivate DB

```

Dieser Befehl ist threadsicher.

Diese Zählerbefehle setzen den aktuellen Wert des benannten Zählers auf seine definierte Mindestzahl zurück.

Optionen

Gibt den Namen des benannten Zählers an, der auf seinen Mindestwert zurückgesetzt werden soll. Der Name kann bis zu 16 alphanumerische Zeichen umfassen. Falls es sich bei *name* um eine Variable handelt, die einen weniger als 16 Zeichen langen Namen enthält, muss der Name mit nachgestellten Leerzeichen aufgefüllt werden.

Gibt den Namen des benannten Zählers an, der auf seinen Mindestwert zurückgesetzt werden soll. Der Name kann bis zu 16 alphanumerische Zeichen umfassen. Falls es sich bei *name* um eine Variable handelt, die einen weniger als 16 Zeichen langen Namen enthält, muss der Name mit nachgestellten Leerzeichen aufgefüllt werden.

Gibt als Vollwort-Binärwert mit Vorzeichen (oder bei DOUNTER als Doppelwort-Binärwert ohne Vorzeichen) das Inkrement an, mit dessen Hilfe ermittelt werden soll, ob der benannte Zähler einen gültigen Status für den Rücksetzvorgang aufweist. Falls ein vorheriger Befehl GET (bei dem nicht die Option REDUCE angegeben war) ein Inkrement angegeben hat, das ein Fehlschlagen des Befehls GET verursachte, geben Sie dasselbe Inkrement im Befehl REWIND an. Der Server für benannte Zähler wendet das Inkrement an, bevor getestet wird, ob der Zähler seinen Grenzwert erreicht hat.

Weitere Details enthält der Abschnitt über die Option INCREMENT für den Befehl GET.

Gibt eine sofortige Rückkehr zum Anwendungsprogramm während einer Neuerstellung der Coupling-Facility-Struktur an. Wenn Sie den Parameter NOSUSPEND nicht angeben, wird die Anforderung so lange wiederholt, bis sie erfolgreich ist. Wenn Sie NOSUSPEND angeben, schlägt die Anforderung sofort fehl und CICS gibt die Bedingung BUSY, RESP2=500 zurück.

Gibt eine 8 Zeichen lange Zeichenfolge an, die als Poolauswahlparameter verwendet werden soll, um den Pool auszuwählen, in dem sich der benannte Zähler befindet. Die Zeichenfolge kann ein logischer Poolname oder der tatsächliche Poolname sein.

Gültige Zeichen für die Poolselektorzichenfolge sind A bis Z, 0 bis 9, \$ @ # und _ (Unterstreichungszeichen). Falls es sich bei *name* um eine Variable handelt, die einen weniger als 8 Zeichen langen Namen enthält, muss der Name mit nachgestellten Leerzeichen aufgefüllt werden.

Dieser Parameter ist optional. Wenn Sie den Namen des Pools nicht angeben, wird ein Poolauswahlwert aus 8 Leerzeichen angenommen.

Wenn kein entsprechender Eintrag in der Optionstabelle DFHNCOPT vorhanden ist, verwendet CICS den Namen, der im Systeminitialisierungsparameter NCPLDFT angegeben ist, der den Standardpool für benannte Zähler angibt.

Informationen zum Generieren einer Optionstabelle für benannte Zähler mithilfe des Makros DFHNCO finden Sie unter Server für benannte Zähler.

Bedingungen

128 BUSY

RESP2-Werte:

500

Die Option NOSUSPEND wurde im Befehl angegeben und die Coupling-Facility-Struktur ist momentan während einer Neuerstellung nicht verfügbar.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

RESP2-Werte:

201

Der benannte Zähler wurde nicht gefunden.

301

Der Server hat einen Fehlercode gemeldet, der für die Schnittstelle des benannten Zählers unverständlich ist. Dies ist in der Regel nur dann möglich, wenn das Schnittstellenlademodul DFHNCIF eine geringere Wartungsstufe oder ein niedrigeres Release-Level als der Server selbst aufweist.

303

Bei einem Makro, das für den Zugriff auf die Coupling-Facility verwendet wird, trat ein unerwarteter Fehler wie beispielsweise eine Strukturstörung oder eine Verbindungsunterbrechung auf. Weitere Informationen enthält die Nachricht DFHNC0441 im Jobprotokoll der Anwendung.

304

Der im Programm angegebene Parameter für die Poolauswahl kann anhand der aktuellen Optionstabelle nicht in einen gültigen Servernamen aufgelöst werden.

305

Die Schnittstelle kann keine Verbindung zum Server für den ausgewählten Pool mit benannten Zählern herstellen. Weitere Informationen enthält eine AXM-Servicenachricht (AXMSCnnnn) im Jobprotokoll der Anwendung.

306

Während der Server eine Anforderung verarbeitete, trat eine abnormale Beendigung auf. Weitere Informationen enthält eine Nachricht im Jobprotokoll der Anwendung und im Jobprotokoll des Servers.

308

Das Optionstabellenmodul DFHNCOPT, das zum Auflösen eines Poolnamens benötigt wird, konnte nicht geladen werden.

309

Während der Verarbeitung der Optionstabelle hat die Schnittstelle für benannte Zähler ein unbekanntes Eintragsformat festgestellt. Entweder wurde die Optionstabelle nicht ordnungsgemäß generiert oder das Schnittstellenlademodul DFHNCIF hat nicht dasselbe Release-Level wie die Optionstabelle.

310

Ein Optionstabelleneintrag, der mit dem angegebenen Poolnamen übereinstimmt, hat ein Benutzerexitprogramm angegeben, aber das Benutzerexitprogramm ist nicht durch eine Linkbearbeitung mit der Optionstabelle verbunden und kann nicht geladen werden.

311

Eine Antwort vom Server des benannten Zählers an das Schnittstellenmodul DFHNCIF der Clientregion gibt an, dass gegenwärtig ein systemverwalteter erneuter Build ausgeführt wird, aber die

Bedingung wird von der EXEC CICS-Schnittstelle nicht erkannt. Dies bedeutet, dass die CICS-Region den Stand von CICS TS 2.1 oder früher aufweist.

403

Der Parameter POOL enthält ungültige Zeichen oder eingebettete Leerzeichen.

404

Der Parameter COUNTER enthält ungültige Zeichen oder eingebettete Leerzeichen.

406

Der Wert für INCREMENT ist ungültig. Der angegebene Wert kann nicht größer als der Gesamtbereich des Zählers ((Maximalwert - Minimalwert) + 1) sein.

Standardaktion: Die Task wird abnormal beendet.

72 SUPPRESSED

RESP2-Werte:

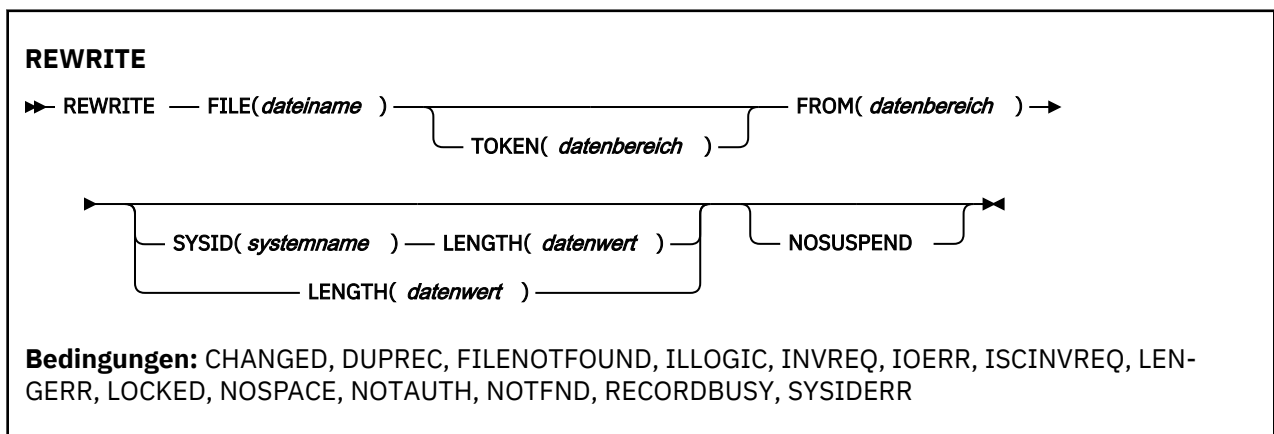
102

Der benannte Zähler hat seinen Grenzwert noch nicht erreicht (der aktuelle Wert ist also nicht gleich dem Maximalwert + 1, was der Bedingung entspricht, dass sich der Zähler an seinem Grenzwert befindet). Diese Fehlerbedingung wird zurückgegeben, falls der benannte Zähler auch nach der Anwendung eines angegebenen Inkrements seinen Grenzwert noch nicht erreicht hat.

Standardaktion: Die Task wird abnormal beendet.

REWRITE

Aktualisiert einen Datensatz in einer Datei.



Dieser Befehl ist threadsicher, wenn für die Datei, auf die er sich bezieht, Folgendes zutrifft:

- Sie ist als fern definiert und der Befehl wird über eine IPIC-Verbindung durch Funktionsverlagerung an eine ferne CICS-Region übertragen.
- Sie ist als lokale VSAM-Datei, als VSAM-RLS-Datei oder als Coupling-Facility-Datentabelle definiert.

Dieser Befehl ist nicht threadsicher, wenn für die Datei, auf die er sich bezieht, Folgendes zutrifft:

- Sie ist als fern definiert und der Befehl wird über eine Nicht-IPIC-Verbindung durch Funktionsverlagerung übertragen.
- Sie ist als gemeinsam genutzte Datentabelle oder als BDAM-Datei definiert.

Beschreibung

Der Befehl REWRITE aktualisiert einen Datensatz in einer Datei auf einem lokalen oder fernen System. Diesem Befehl muss stets ein Lesevorgang mit der Option UPDATE vorausgehen.

Bei VSAM-Dateien dürfen Sie das Schlüsselfeld im Datensatz nicht ändern.

Wenn dieser Befehl eingesetzt wird, um einen Datensatz in einer von CICS verwalteten Datentabelle zu ändern, wird die Aktualisierung sowohl an der schlüsseladressierten VSAM-Quellendatei als auch an der speicherinternen Datentabelle vorgenommen. Die Details des Befehls für eine von CICS verwaltete Tabelle sind dieselben wie für eine schlüsseladressierte VSAM-Datei.

Wenn dieser Befehl eingesetzt wird, um einen Datensatz in einer benutzerverwalteten Datentabelle zu ändern, wird die Aktualisierung an der speicherinternen Datentabelle vorgenommen.

Wenn dieser Befehl eingesetzt wird, um Datensätze in einer Coupling-Facility-Datentabelle zu ändern, wird ausschließlich die Datentabelle in der Coupling-Facility aktualisiert.

Falls die Aktualisierung einer benutzerverwalteten oder Coupling-Facility-Datentabelle aufgrund fehlenden Speicherbereichs erfolglos ist, wird an den Befehl die Antwort NOSPACE zurückgegeben. CICS gibt außerdem die Nachricht DFHFC0432 aus, um Systemadministratoren über die Situation zu benachrichtigen.

Optionen

FILE(dateiname)

Gibt den Namen der Datei an, auf die zugegriffen werden soll.

Bei Angabe eines Wertes für SYSID wird davon ausgegangen, dass sich die Datei, auf die sich dieser Dateiname bezieht, auf einem fernen System befindet, wobei es ohne Belang ist, ob der Name für CICS definiert ist. Ansonsten wird die Ressourcendefinition verwendet, um zu ermitteln, ob sich die Datei (Dataset) auf einem lokalen oder fernen System befindet.

FROM(datenbereich)

Gibt den Datensatz an, der in die durch diesen Namen angegebene Datei geschrieben werden soll.

LENGTH(datenwert)

Gibt als Halbwort-Binärwert die Länge des Datenbereichs an, aus dem der Datensatz geschrieben werden soll.

Bei Angabe von SYSID muss diese Option angegeben werden.

Falls sich die Datei auf einem fernen System befindet, muss der Parameter LENGTH hier nicht festgelegt werden, aber in der FILE-Ressourcendefinition festgelegt sein.

Falls sich die Datei auf einem lokalen System befindet, muss der Parameter LENGTH mithilfe der Option INTO für Datensätze mit variabler Länge festgelegt werden, jedoch nicht für Datensätze mit fester Länge. Die Angabe der Länge für Datensätze mit fester Länge ist jedoch ratsam, weil hierdurch geprüft wird, dass die Länge des zu schreibenden Datensatzes die für die Datei geltende Definition nicht überschreitet.

NOSUSPEND (nur RLS)

Die Anforderung wartet nicht, falls VSAM eine aktive Sperre für den Datensatz hält (hierzu gehören auch Datensätze, die infolge einer Bedingung DEADLOCK gesperrt sind).

Ein Sperrenkonflikt kann auftreten, falls die Aktualisierung Änderungen einbezieht, die im RLS-Modus an Datensätzen in einer VSAM-Datei vorgenommen werden, für die es einen oder mehrere Alternativindizes gibt, und ein Alternativindex mit eindeutigen Schlüsseln definiert ist.

In den meisten anderen Fällen sollten Sie diese Option nicht benötigen, weil die aktive Sperre angefordert wird, sobald die Task den Befehl **READ UPDATE** ausgibt.

Anmerkung: Anforderungen mit der Angabe NOSUSPEND warten mindestens eine Sekunde, bevor CICS die Antwort RECORDBUSY zurückgibt.

SYSID(systemname)

Gibt den Namen des Systems an, an das die Anforderung gerichtet ist.

TOKEN(datenbereich)

Gibt in Form eines Vollwort-Binärwertes eine eindeutige Kennung für einen Befehl REWRITE an, die den Befehl einem vorherigen Befehl READ, READNEXT oder READPREV zuordnet, bei dem die Option UPDATE angegeben war.

TOKEN kann durch Funktionsverlagerung übertragen werden. Falls jedoch eine Anforderung mit Angabe von TOKEN per Funktionsverlagerung an ein Mitglied der CICS-Produktfamilie übertragen wird, das die Option TOKEN nicht unterstützt, schlägt die Anforderung fehl.

Bedingungen

105 CHANGED

RESP2-Werte:

109

Ein Befehl REWRITE wurde für eine Datei ausgegeben, die als Coupling-Facility-Datentabelle definiert ist, die das Aktualisierungsmodell für Konflikte verwendet und für die der Datensatz geändert wurde, seitdem er vom Anwendungsprogramm für die Aktualisierung eingelesen wurde. Zur erfolgreichen Aktualisierung des Datensatzes muss der Befehl READ für die Option UPDATE wiederholt werden, damit die neueste Version des Datensatzes abgerufen wird. Anschließend muss die Änderung angewendet und der Befehl REWRITE wiederholt werden.

Standardaktion: Die Task wird abnormal beendet.

14 DUPREC

RESP2-Werte:

150

Es wurde versucht, einen Datensatz in einer Datei neu zu schreiben, deren Upgradegruppe einen Alternativindex mit dem Attribut UNIQUEKEY besitzt, falls der entsprechende alternative Schlüssel bereits im Alternativindex vorhanden ist.

Standardaktion: Die Task wird abnormal beendet.

12 FILENOTFOUND

RESP2-Werte:

1

Ein in der Option FILE referenzierter Dateiname ist nicht für CICS definiert.

Standardaktion: Die Task wird abnormal beendet.

21 ILLOGIC

Jeder gegebenenfalls gerade ausgeführte Suchvorgang wird beendet, wenn diese Bedingung auftritt.

RESP2-Werte (VSAM):

110

Es ist ein VSAM-Fehler aufgetreten, der zu keiner der anderen CICS-Antwortkategorien gehört.

Weitere Informationen sind bei EIBRCODE im EXEC-Schnittstellenblock verfügbar; Details finden Sie unter [EIB-Felder](#).

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

RESP2-Werte:

30

Ein Befehl REWRITE wurde ohne ein Token ausgegeben und es wurde kein vorheriger Befehl READ für die Option UPDATE (ebenfalls ohne Token) gefunden.

Möglicherweise wurde der vorherige Befehl READ für die Option UPDATE nicht gefunden, weil er aus nicht näher bekanntem Grund fehlgeschlagen ist und das Fehlschlagen entweder nicht ordnungsgemäß bearbeitet oder aber ignoriert wurde.

46

Ein Befehl REWRITE hat versucht, die Länge eines BDAM-Datensatzes oder -Blocks mit variabler Länge zu ändern.

47

Eine Anweisung REWRITE enthielt ein Token, dessen Wert mit keinem der Tokens übereinstimmt, die bei einer vorhandenen Anforderung von READ für die Option UPDATE verwendet werden.

55

Die Leseintegrität NOSUSPEND ist nicht zulässig, weil es sich nicht um eine VSAM-Datei handelt, auf die im RLS-Modus zugegriffen wird.

56

Der Versuch, eine wiederherstellbare Coupling-Facility-Datentabelle zu aktualisieren, ist fehlgeschlagen, weil die aktuelle Arbeitseinheit bereits 1024 wiederherstellbare Coupling-Facility-Datentabellen aktualisiert hat. Innerhalb einer einzigen Arbeitseinheit können nicht mehr als 1024 wiederherstellbare Coupling-Facility-Datentabellen aktualisiert werden.

57

Operationen REWRITE sind nicht zulässig, weil die zugehörige VSAM-Datei für den Status AVAILABILITY den Wert RREPL aufweist, der solche Operationen ausschließlich durch Replikationsprogramme zulässt.

Standardaktion: Die Task wird abnormal beendet.

17 IOERR

RESP2-Werte:

120

Während der Dateisteuerungsoperation trat ein E/A-Fehler auf. Ein E/A-Fehler ist ein nicht erwartetes Ereignis, das nicht durch eine CICS-Bedingung abgedeckt ist.

Bei VSAM-Dateien gibt die Bedingung IOERR für gewöhnlich einen Hardwarefehler an. Weitere Informationen sind im EXEC-Schnittstellenblock verfügbar; Details finden Sie unter [EIB-Felder](#).

Bei einer Coupling-Facility-Datentabelle gibt eine Bedingung IOERR an, dass eine falsche Antwort von einem Coupling-Facility-Zugriff zurückgegeben wurde.

Standardaktion: Die Task wird abnormal beendet.

54 ISCINVREQ

RESP2-Werte:

70

Das ferne System gibt einen Fehler an, der keiner bekannten Bedingung entspricht.

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

RESP2-Werte:

10

Die Option LENGTH war für eine Datei mit Datensätzen variabler Länge oder für eine BDAM-Datei mit Datensätzen nicht definierten Formats nicht angegeben.

12

Die angegebene Länge überschreitet die maximale Datensatzgröße (der Quellendatei für eine Datentabelle); der Datensatz wird abgeschnitten.

14

Für eine Datei mit Datensätzen fester Länge wurde eine falsche Länge angegeben.

Standardaktion: Die Task wird abnormal beendet.

13 NOTFND

RESP2-Werte:

80

Bei benutzerverwalteten Datentabellen tritt diese Bedingung auf, wenn ein versuchter Befehl REWRITE für einen Datensatz fehlgeschlagen ist, weil der Befehl REWRITE einer Anforderung READ UPDATE für einen Datensatz zugeordnet ist, der von dieser Transaktion (durch DELETE mit Option RIDFLD) gelöscht wurde, nachdem er für die Aktualisierung eingelesen wurde. Dies kann auf einen Logikfehler im Anwendungsprogramm zurückzuführen sein.

Diese Bedingung kann außerdem auftreten, wenn ein Befehl REWRITE für eine Coupling-Facility-Datentabelle ausgegeben wird, die das Konfliktmodell verwendet, und der Datensatz gelöscht wurde, nachdem er für die Aktualisierung eingelesen worden ist.

Standardaktion: Die Task wird abnormal beendet.

100 LOCKED

RESP2-Werte:

106

Es wurde versucht, einen Datensatz zu aktualisieren, aber für einen eindeutigen Alternativschlüssel, der in die Anforderung einbezogen ist, besteht eine beibehaltene Sperre.

Standardaktion: Die Task wird mit dem Code AEX8 abnormal beendet.

18 NOSPACE

RESP2-Werte:

100

Auf der Direktzugriffseinheit ist kein Speicherbereich für das Hinzufügen des aktualisierten Datensatzes zur Datei verfügbar.

102

Die maximale Anzahl von Datensätzen, die für eine wiederherstellbare Coupling-Facility-Datentabelle angegeben ist, wurde erreicht. Dies kann bei einer Operation REWRITE auftreten, weil zu Wiederherstellungszwecken in der Coupling-Facility-Datentabelle ein zusätzlicher Datensatz benötigt wird, bis die Aktualisierung festgeschrieben worden ist.

103

Diese Bedingung tritt bei benutzerverwalteten Datentabellen auf, falls CICS nicht genügend Speicher im CICS-Adressraum abrufen kann, um den aktualisierten Datentabelleneintrag zu speichern.

108

Diese Bedingung tritt bei Coupling-Facility-Datentabellen auf, falls der Pool für Coupling-Facility-Datentabellen nicht genügend Speicherbereich zum Speichern des aktualisierten Datensatzes enthält.

Standardaktion: Die Task wird abnormal beendet.

70 NOTAUTH

RESP2-Werte:

101

Eine Ressourcensicherheitsprüfung ist für FILE(dateiname) fehlgeschlagen.

Standardaktion: Die Task wird abnormal beendet.

101 RECORDBUSY

RESP2-Werte:

107

Die Leseintegrität NOSUSPEND ist angegeben, aber VSAM hält eine aktive Sperre für einen eindeutigen Alternativindexschlüssel, der in die Anforderung einbezogen ist, was einen Wartestatus für die Anforderung zur Folge hätte. Weitere Informationen enthält der Abschnitt Beibehaltene und aktive Sperren.

Standardaktion: Die Task wird mit dem Code AEX9 abnormal beendet.

53 SYSIDERR

RESP2-Werte:

130

Die Option SYSID gibt einen Namen an, bei dem es sich weder um das lokale System noch um ein fernes System handelt (das CICS durch eine CONNECTION-Definition bekanntgegeben wird). Die Bedingung SYSIDERR tritt außerdem auf, wenn die Verbindung zu dem fernen System geschlossen ist.

131

Für eine Coupling-Facility-Datentabelle ist die Verbindung zum Server für Coupling-Facility-Datentabellen fehlgeschlagen. Dies könnte dadurch verursacht worden sein, dass der Server selbst fehlgeschlagen ist, oder dass der Server zwar verfügbar ist, CICS jedoch keine Verbindung zu ihm herstellen konnte.

132

Der Befehl REWRITE wurde für eine Coupling-Facility-Datentabelle ausgegeben, die nicht mehr vorhanden ist. Vermutlich liegt dies an einem Fehler der Coupling-Facility, durch den der Server für Coupling-Facility-Datentabellen ebenfalls fehlschlägt. Informationen zum erneuten Starten eines Servers für Coupling-Facility-Datentabellen und zum erneuten Laden einer Tabelle finden Sie unter [Server für Coupling-Facility-Datentabellen einrichten und ausführen](#).

Standardaktion: Die Task wird abnormal beendet.

Beibehaltene und aktive Sperren: RECORDBUSY bezieht sich auf aktive Sperren, LOCKED bezieht sich auf beibehaltene Sperren:

- Anforderungen REWRITE für Datensätze mit *beibehaltenen* Sperren werden immer mit einer Antwort LOCKED zurückgewiesen.
- Anforderungen REWRITE für Datensätze mit *aktiven* Sperren warten auf die Freigabe der Sperre. Ausgenommen sind Fälle, bei denen das Schlüsselwort NOSUSPEND angegeben ist; hier gibt CICS die Antwort RECORDBUSY zurück.

Beispiele

Das folgende Beispiel zeigt einen einfachen Befehl REWRITE:

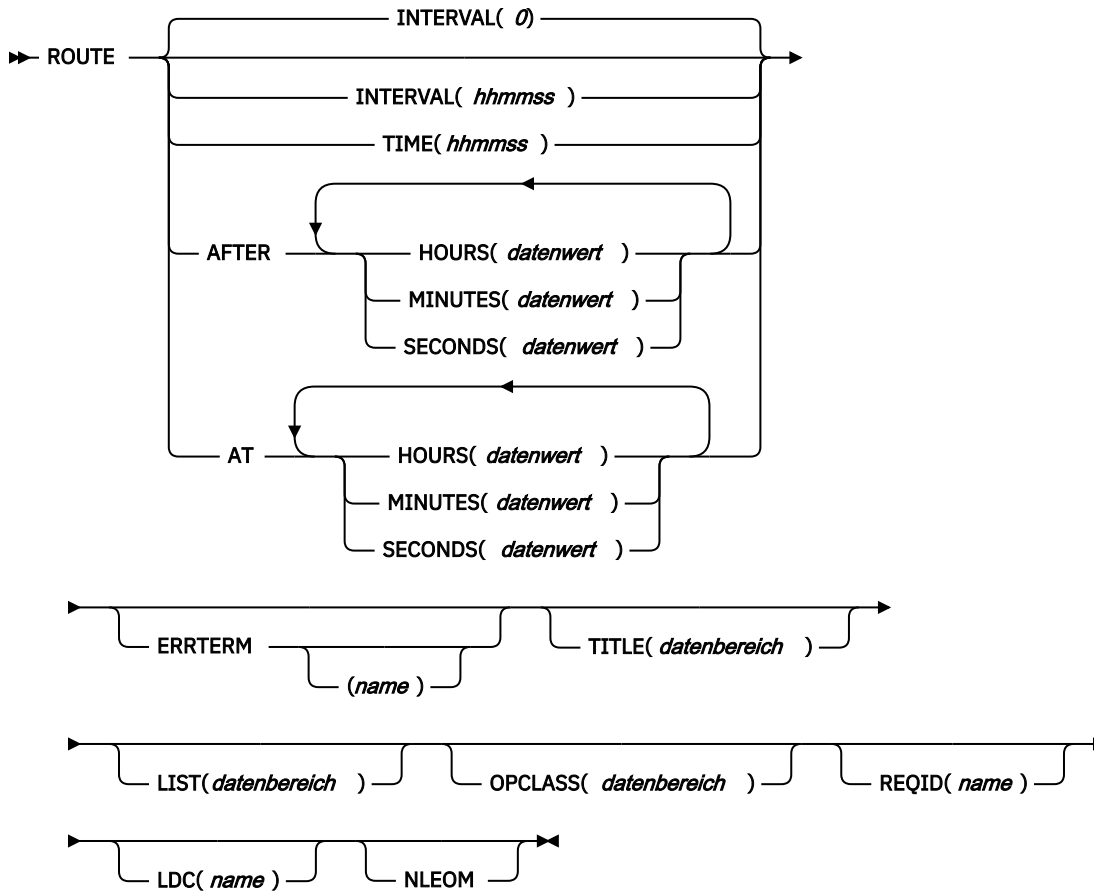
```
EXEC CICS REWRITE  
      FROM(RECORD)  
      FILE('MASTER')  
      TOKEN(APTOK)
```

ROUTE

Leitet eine BMS-Nachricht weiter. Dieser Befehl ist nur bei vollständiger BMS verfügbar.

Weitere Informationen zu BMS finden Sie unter [Basic Mapping Support](#).

ROUTE



Bedingungen: IGRREQID, INVERRTERM, INVLDC, INVREQ, RTEFAIL, RTESOME

Beschreibung

Der Befehl **ROUTE** leitet eine logische BMS-Nachricht an eines oder mehrere Terminals bzw. einen oder mehrere Terminalbediener weiter.

Der Standardwert ist **INTERVAL(0)**; bei C lautet der Standardwert jedoch **AFTER HOURS(0) MINUTES(0) SECONDS(0)**.

Optionen

AFTER

Gibt den Zeitraum an, der vor der Weiterleitung verstreichen soll.

Für die Angabe der Zeit unter den Optionen **AFTER** und **AT** gibt es zwei Möglichkeiten:

1. Als Kombination von mindestens zwei der Optionen **HOURS(0-99)**, **MINUTES(0-59)** und **SECONDS(0-59)**. Zum Beispiel würde die Angabe **HOURS(1) SECONDS(3)** eine Stunde und drei Sekunden bedeuten (für die Minuten wird standardmäßig null angenommen).
2. Durch einen der Werte **HOURS(0-99)**, **MINUTES(0-5999)** oder **SECONDS(0-359 999)**. Die Angabe **HOURS(1)** bedeutet 1 Stunde. **MINUTES(62)** bedeutet eine Stunde und zwei Minuten. Die Angabe **SECONDS(3723)** bedeutet 1 Stunde, 2 Minuten und 3 Sekunden.

AT

Gibt den Zeitpunkt für die Weiterleitung an. Informationen zu den Möglichkeiten der Zeitangabe finden Sie unter der Option **AFTER**.

ERRTERM(name)

Gibt den Namen des Terminals an, das benachrichtigt werden soll, falls die Nachricht gelöscht wird, weil sie unzustellbar ist. Hierzu werden die Nachrichtennummer, die Titeltkennzeichnung und das Ziel angegeben. Falls kein Name angegeben wird, wird das Ursprungsterminal angenommen.

Diese Option ist nur dann wirksam, wenn in den Systeminitialisierungsparametern PRGDLY angegeben wurde.

HOURS(datenwert)

Gibt einen Vollwort-Binärwert im Bereich von 0 bis 99 an. Dies ist eine Unteroption der Optionen AFTER und AT. Informationen zur Verwendung und zur Bedeutung finden Sie unter der Option AFTER.

INTERVAL(hhmmss)

Gibt das Zeitintervall an, nach dem die Daten an die im Befehl ROUTE angegebenen Terminals übertragen werden sollen. Die Werte **mm** und **ss** liegen im Bereich von 0 bis 59.

Bei Verwendung der Programmiersprache C empfiehlt sich die Verwendung der Optionen AFTER/AT mit den Optionen HOURS, MINUTES und SECONDS, da C den Datentyp für gepackte Dezimalzahlen nicht unterstützt. Sie können die Option INTERVAL zwar verwenden, aber falls es sich bei dem angegebenen Wert **nicht** um eine ganzzahlige Konstante handelt, muss die Anwendung sicherstellen, dass der an CICS übergebene Wert ein gepacktes Dezimalformat besitzt.

LDC(name) (nur für logische Einheiten)

Gibt ein zweistelliges mnemonisches Zeichen an, mit dessen Hilfe der logische Einheitencode (Logical Device Code, LDC) ermittelt wird, der im Funktionsverwaltungshdr an die logische Einheit übertragen werden soll. Das mnemonische Zeichen gibt einen LDC-Eintrag an, der durch das Makro DFHTCT TYPE=LDC definiert wird.

Bei Angabe eines logischen Einheitencodes verwendet BMS den Einheitentyp, die Seitengröße und den Seitenstatus, die dem logischen Einheitencode zugeordnet sind, um die Nachricht zu formatieren. Diese Werte werden aus der erweiterten lokalen LDC-Tabelle für die logische Einheit entnommen, falls sie eine solche Tabelle besitzt. Für den Fall, dass die logische Einheit nur eine lokale (nicht erweiterte) LDC-Tabelle besitzt, werden die Werte aus der LDC-Tabelle des Systems entnommen. Der numerische Wert des logischen Einheitencodes wird aus der lokalen LDC-Tabelle abgerufen, es sei denn, dass dies eine nicht erweiterte Tabelle ist und der Wert nicht angegeben ist. In diesem Fall wird er aus der Systemtabelle abgerufen.

Falls die Option LDC nicht angegeben ist, wird das in DFHMSD angegebene mnemonische Zeichen für den logischen Einheitencode verwendet; weitere Details enthält der Abschnitt „DFHMSD“ auf Seite 871. Wenn die Option LDC auch in DFHMSD nicht angegeben ist, richtet sich die Aktion wie folgt nach dem Typ der logischen Einheit:

Logische 3601-Einheit

Es wird der erste Eintrag in einer lokalen oder einer erweiterten lokalen LDC-Tabelle verwendet, wenn es eine solche Tabelle gibt. Falls auf diese Weise kein Standardwert abgerufen werden kann, wird der numerische Wert für einen leeren LDC (X'00') verwendet. Als Seitengröße wird der Wert verwendet, der durch die RDO TYPETERM-Optionen PAGESIZE oder ALTPAGE angegeben ist, bzw. der Wert (1,40), falls diese Optionen keinen Wert angeben.

Logische Einheit mit LU-Typ 4, mit Stapel oder mit Stapeldatenaustausch

Es wird nicht die lokale LDC-Tabelle für die Bereitstellung eines Standard-LDC verwendet. Stattdessen wird die Nachricht an die Konsole der logischen Einheit übertragen. (Als Konsole der logischen Einheit gilt in diesem Zusammenhang jedes Medium, das die logische Einheit für den Empfang solcher Nachrichten wählt. Bei einer logischen Einheit mit Stapeldatenaustausch impliziert dies nicht das Senden eines logischen Einheitencodes im Funktionsverwaltungshdr.) Die Seitengröße wird auf dieselbe Weise wie bei einer logischen 3601-Einheit abgerufen.

Bei der Nachrichtenweiterleitung hat die Option LDC des Befehls ROUTE Vorrang vor allen anderen Quellen. Wenn diese Option weggelassen wird und eine Routing-Liste angegeben ist (Option LIST), wird das mnemonische Zeichen für den logischen Einheitencode aus der Routing-Liste verwendet. Falls die Routing-Liste kein solches mnemonisches Zeichen enthält oder keine Routing-Liste angegeben ist, wird für den logischen Einheitencode wie oben beschrieben ein Standardwert ausgewählt.

LIST(datenbereich)

Gibt den Datenbereich an, der eine Liste von Terminals und Bedienern enthält, an die die Daten übertragen werden sollen. Wenn diese Option weggelassen wird, empfangen alle durch BMS unterstützten Terminals die Daten (sofern nicht die Option OPCLASS wirksam ist). Das Format der Routing-Liste ist unter Format der Routing-Liste erläutert.

MINUTES(datenwert)

Gibt einen Vollwort-Binärwert aus dem Bereich von 0 bis 59 an, wenn auch die Optionen HOURS oder SECONDS angegeben werden, oder gibt einen Wert aus dem Bereich von 0 bis 5999 an, wenn MINUTES die einzige Option ist, die angegeben wird. Dies ist eine Unteroption der Optionen AFTER und AT. Informationen zur Verwendung und zur Bedeutung finden Sie unter der Option AFTER.

NLEOM

Gibt an, dass Daten für einen 3270-Drucker oder eine 3275-Anzeige mit Druckeradapterfunktion mit Leerzeichen und Zeilenvorschubzeichen erstellt werden sollen und dass am Ende der Daten ein Zeichen für das Nachrichtenende eingefügt werden soll. Beim Drucken der Daten bewirkt jedes Zeilenvorschubzeichen, dass die Druckausgabe in der nächsten Zeile fortgesetzt wird, und das Zeichen für das Nachrichtenende beendet die Druckausgabe.

Die Option wird ignoriert, falls die Einheit, die die Nachricht (direkt oder weitergeleitet) empfängt, keine der oben genannten Einheiten ist.

Falls diese Option verwendet wird, ist weder eine Pufferaktualisierung noch eine Attributänderung von zuvor in den Puffer geschriebenen Feldern zulässig. CICS bezieht bei jedem Schreibvorgang für das Terminal die Option ERASE ein.

Das Zeilenvorschubzeichen belegt eine Pufferposition. Eine Reihe von Pufferpositionen, deren Anzahl dem Wert der RDO-Optionen PAGESIZE oder ALTPAGE für dieses Terminal entspricht, sind für Daten nicht verfügbar. Dies kann dazu führen, dass Daten im Puffer umlaufen; in einem solchen Fall muss der Wert für PAGESIZE oder ALTPAGE herabgesetzt werden.

OPCLASS(datenbereich)

Gibt den Datenbereich an, der eine Liste von Bedienerklassen enthält, an die die Daten weitergeleitet werden sollen. Die Klassen werden in einem 3 Byte umfassenden Feld angegeben, wobei jede Bitposition einem der Codes im Bereich von 1 bis 24 entspricht, jedoch in umgekehrter Reihenfolge. Dies bedeutet, dass das erste Byte den Codes 24 bis 17, das zweite Byte den Codes 16 bis 9 und das dritte Byte den Codes 8 bis 1 entspricht.

REQID(name)

Gibt in einem zweistelligen Feld ein Präfix an, das als Teil einer Kennung des temporären Speichers für die CICS-Nachrichtenwiederherstellung verwendet werden soll. Für jede logische Nachricht kann nur ein einziges Präfix angegeben werden. Das Standardpräfix ist **.

Die BMS-Nachrichtenwiederherstellung steht für eine logische Nachricht nur dann zur Verfügung, wenn die Option PAGING in den BMS-Befehlen SEND angegeben ist und wenn der Synchronisationspunkt erreicht wurde.

SECONDS(datenwert)

Gibt einen Vollwort-Binärwert aus dem Bereich von 0 bis 59 an, wenn auch die Optionen HOURS oder MINUTES angegeben werden, oder gibt einen Wert aus dem Bereich von 0 bis 359 999, wenn SECONDS die einzige Option ist, die angegeben wird. Dies ist eine Unteroption der Optionen AFTER und AT. Informationen zur Verwendung und zur Bedeutung finden Sie unter der Option AFTER.

TIME(hhmmss)

Gibt die Uhrzeit an, zu der die Daten an die im Befehl ROUTE angegebenen Terminals übertragen werden sollen.

Bei Verwendung der Programmiersprache C empfiehlt sich die Verwendung der Optionen AFTER/AT mit den Optionen HOURS, MINUTES und SECONDS, da C den Datentyp für gepackte Dezimalzahlen nicht unterstützt. Sie können die Option TIME zwar verwenden, aber falls es sich bei dem angegebenen Wert **nicht** um eine ganzzahlige Konstante handelt, muss die Anwendung sicherstellen, dass der an CICS übergebene Wert ein gepacktes Dezimalformat besitzt.

TITLE(datenbereich)

Gibt den Datenbereich an, in dem der Titel enthalten ist, der für eine weitergeleitete logische Nachricht verwendet werden soll. Dieser Titel wird als Teil der Antwort auf einen Seitenabfragebefehl ausgegeben. Das Format der Option TITLE ist unter Aspekte der Programmierung mit Routing erläutert.

Bedingungen**39 IREQID**

Tritt auf, falls das in der Option REQID angegebene Präfix nicht mit dem von einer vorherigen Option REQID oder mit dem standardmäßig für diese logische Nachricht erstellten Präfix, also REQID (**), identisch ist.

37 INVERRTERM

Tritt auf, falls die in der Option ERRTERM angegebene Terminalkennung ungültig oder einem Terminaltyp zugeordnet ist, der von BMS nicht unterstützt wird.

Standardaktion: Die Task wird abnormal beendet.

41 INVLDC

Tritt auf, falls das angegebene mnemonische Zeichen für den logischen Einheitencode (LDC) nicht in der LDC-Liste für die logische Einheit enthalten ist.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

RESP2-Werte:

4

Die Angabe für die Stunden liegt außerhalb des gültigen Bereichs.

5

Die Angabe für die Minuten liegt außerhalb des gültigen Bereichs.

6

Die Angabe für die Sekunden liegt außerhalb des gültigen Bereichs.

200

BMS-Befehle werden für Distributed Program Link nicht unterstützt.

Diese Bedingung (RESP2 nicht festgelegt) tritt auch in den folgenden Situationen auf:

- Die Byte 10 bis 15 eines Routing-Listeneintrags enthalten keine Leerzeichen.

Standardaktion: Die Task wird abnormal beendet.

33 RTEFAIL

Tritt in einer der folgenden Situationen auf:

- Ein Befehl ROUTE würde dazu führen, dass die Nachricht ausschließlich an das Terminal gesendet wird, das die Transaktion initialisiert hat.
- Ein Befehl ROUTE wird für ein fernes übertragbares Terminal ausgegeben, das noch nicht in der Application-Ownning Region installiert ist.

Standardaktion: Die Steuerung wird unmittelbar nach dem Befehl ROUTE an das Anwendungsprogramm zurückgegeben.

34 RTE SOME

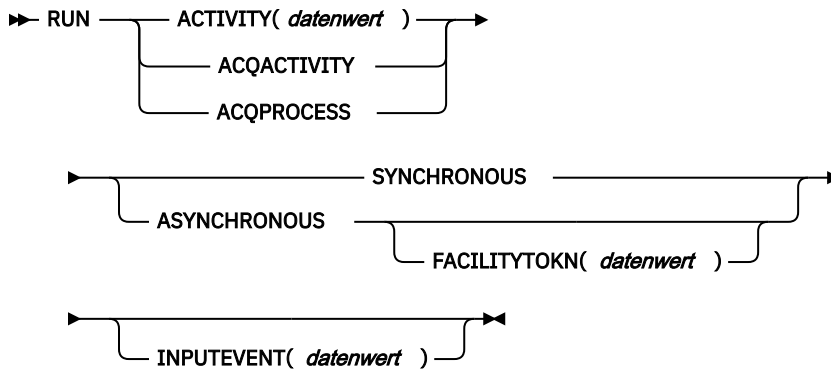
Tritt auf, falls eines der Terminals, die in den Optionen des Befehls ROUTE angegeben sind, die Nachricht nicht empfangen hat.

Standardaktion: Die Steuerung wird unmittelbar nach dem Befehl ROUTE an das Anwendungsprogramm zurückgegeben.

RUN

Führt einen CICS-BTS-Prozess oder eine BTS-Aktivität synchron oder asynchron mit Kontextwechsel aus.

RUN



Bedingungen: ACTIVITYBUSY, ACTIVITYERR, EVENTERR, INVREQ, IOERR, LOCKED, NOTAUTH, PROCESSBUSY, PROCESSERR

Beschreibung

Der Befehl RUN führt einen CICS-BTS-Prozess oder eine CICS-BTS-Aktivität synchron oder asynchron zum Anforderer (Requester) mit Kontextwechsel aus. Der Prozess oder die Aktivität muss zuvor für BTS definiert worden sein.

Der Befehl RUN veranlasst BTS, den Prozess oder die Aktivität erneut anzuschließen (zu reaktivieren), indem ein Eingabeereignis an den Prozess bzw. die Aktivität gesendet wird. Wenn sich der Prozess oder die Aktivität im Anfangsstatus befindet, d. h., wenn dies das erste Mal ist, dass er bzw. sie ausgeführt werden soll oder wenn er bzw. sie durch einen Befehl RESET ACTIVITY zurückgesetzt wurde, sendet CICS das Systemereignis DFHINITIAL an den Prozess bzw. die Aktivität. Wenn der Prozess oder die Aktivität im Ruhemodus (DORMANT) ist, das heißt, wenn er bzw. sie auf ein Reaktivierungsereignis wartet, muss das Eingabeereignis in der Option INPUTEVENT angegeben werden.

Wenn sich der Prozess oder die Aktivität in einem anderen Modus als INITIAL oder DORMANT befindet, kann er bzw. sie nicht ausgeführt werden.

Mithilfe der Optionen SYNCHRONOUS und ASYNCHRONOUS können Sie angeben, ob der Prozess bzw. die Aktivität synchron oder asynchron zum Anforderer ausgeführt werden soll.

Kontextwechsel

Wenn ein Prozess oder eine Aktivität durch einen Befehl RUN aktiviert wird, wird er bzw. sie wie folgt ausgeführt:

- In einer vom Anforderer separaten Arbeitseinheit.
- Mit Angabe der Transaktionsattribute (TRANSID und USERID) im Befehl DEFINE PROCESS oder DEFINE ACTIVITY.

Mit anderen Worten, es findet ein **Kontextwechsel** statt. Die Beziehung des Prozesses oder der Aktivität zum Anforderer ist der Beziehung zwischen separaten Transaktionen ähnlich, jedoch mit folgenden Ausnahmen:

- Daten können zwischen den beiden Arbeitseinheiten übergeben werden.
- Der Start und das Ende der Aktivität hängt mit den Synchronisationspunkten des Anforderers zusammen.

Zum Ausführen eines Prozesses oder einer Aktivität *ohne* Kontextwechsel, d. h. in derselben Arbeitseinheit und mit denselben Attributen TRANSID und USERID wie die anfordernde Transaktion, verwenden Sie den Befehl LINK ACQPROCESS, LINK ACQACTIVITY oder LINK ACTIVITY. Dies ist nur möglich, wenn der Prozess bzw. die Aktivität synchron ausgeführt wird.

Wenn die Möglichkeit, einen Fehler zu isolieren, wichtiger als die Leistung ist, verwenden Sie den Befehl RUN SYNCHRONOUS und nicht den Befehl LINK.

Aktivitäten

Die einzigen Aktivitäten, die ein Programm ausführen kann, sind die folgenden:

- Seine eigenen untergeordneten Aktivitäten, falls es als Aktivierung einer Aktivität ausgeführt wird. Es kann mehrere der eigenen untergeordneten Aktivitäten innerhalb derselben Arbeitseinheit ausführen.
- Die Aktivität, die es durch einen Befehl ACQUIRE ACTIVITYID in der aktuellen Arbeitseinheit angefordert hat.

Zur Prüfung der Antwort von der Aktivität muss der Befehl CHECK ACTIVITY verwendet werden. Dies hat den Grund, dass die Antwort auf die Anforderung zum Ausführen der Aktivität keine Informationen über Erfolg oder Fehler der Aktivität selbst enthält, sondern nur über Erfolg oder Fehler der Anforderung zum Ausführen der Aktivität.

Wenn die Aktivität synchron ausgeführt wird, wird in der Regel der Befehl CHECK sofort nach dem Befehl RUN abgesetzt. Wird sie asynchron ausgeführt, kann der Befehl CHECK unter folgenden Bedingungen abgesetzt werden:

- Wenn die übergeordnete Aktivität der Aktivität erneut angeschlossen (reaktiviert) wurde, weil das Beendigungsereignis der Aktivität ausgelöst wurde.
- Wenn der Anforderer aufgrund des Ablaufs eines Zeitgebers erneut angeschlossen (reaktiviert) wird.

Das Beendigungsereignis der Aktivität ist eines der folgenden:

1. Das Ereignis, das in der Option EVENT des Befehls DEFINE für die Aktivität angegeben wurde.
2. Wenn der Befehl DEFINE kein Beendigungsereignis angegeben hat, ein Ereignis mit demselben Namen wie die Aktivität.

Gehen Sie wie folgt vor, um eine Aktivität erneut zu versuchen:

1. Setzen Sie einen Befehl RESET ACTIVITY ab, um die Aktivität auf Ihren Anfangsstatus (INITIAL) zurückzusetzen.
2. Setzen Sie einen Befehl RUN ab.

Prozesse

Der einzige Prozess, den ein Programm ausführen kann, ist derjenige, den es in der aktuellen Arbeitseinheit angefordert hat (siehe [Prozesse und Aktivitäten anfordern](#)).

Zur Prüfung der Antwort von dem Prozess muss der Befehl CHECK ACQPROCESS verwendet werden. Dies hat den Grund, dass die Antwort auf die Anforderung zum Ausführen des Prozesses keine Informationen über Erfolg oder Fehler des Prozesses selbst enthält, sondern nur über Erfolg oder Fehler der Anforderung zum Ausführen des Prozesses.

Wenn der Prozess synchron ausgeführt wird, wird in der Regel der Befehl CHECK sofort nach dem Befehl RUN abgesetzt. Wird der Prozess asynchron ausgeführt, kann der Befehl CHECK abgesetzt werden, wenn der Anforderer aufgrund des Ablaufs eines Zeitgebers erneut angeschlossen (reaktiviert) wurde.

Optionen

ACQACTIVITY

Gibt an, dass die auszuführende Aktivität diejenige ist, die von der aktuellen Arbeitseinheit durch einen Befehl ACQUIRE ACTIVITYID angefordert wurde.

ACQPROCESS

Gibt an, dass der Prozess, der zurzeit durch den Requester angefordert ist, ausgeführt werden soll.

ACTIVITY(datenwert)

Gibt den Namen (1-16 Zeichen) der Aktivität an, die ausgeführt werden soll. Der Name muss der einer untergeordneten Aktivität der aktuellen Aktivität sein.

ASYNCHRONOUS

Gibt an, dass der Prozess bzw. die Aktivität asynchron zum Requester ausgeführt werden soll.

FACILITYTKN(datenwert)

Gibt ein acht Byte langes Brückeneinrichtungstoken an.

Diese Option gilt, wenn eine BTS-Clientaktivität eine 3270-basierte Pseudodialogtransaktion ausführt. Um sicherzustellen, dass die vorhandene Brückeneinrichtung für die nächste Transaktion im Pseudodialog wiederverwendet wird, übergibt der Client sein Token an die nächste untergeordnete Aktivität. Dies wird unter Vorhandene 3270-Anwendungen in BTS wiederverwenden näher erläutert.

INPUTEVENT(datenwert)

Gibt den Namen (1-16 Zeichen) des Ereignisses an, das veranlasst, dass der Prozess oder die Aktivität angeschlossen wird.

Sie dürfen diese Option *nicht* angeben, wenn sich der Prozess oder die Aktivität im Anfangsstatus (INITIAL) befindet, d. h., wenn dies das erste Mal ist, dass er bzw. sie ausgeführt werden soll, oder wenn die Aktivität durch einen Befehl RESET ACTIVITY zurückgesetzt wurde. In diesem Fall sendet CICS dem Prozess oder der Aktivität das Systemereignis DFHINITIAL.

Sie *müssen* diese Option angeben, wenn sich der Prozess oder die Aktivität nicht im Anfangsstatus befindet, d. h. wenn er bzw. sie zuvor bereits aktiviert wurde und die Aktivität nicht durch einen Befehl RESET ACTIVITY zurückgesetzt wurde.

Wenn Sie die Option INPUTEVENT angeben, muss für eine erfolgreiche Ausführung des Befehls RUN das angegebene Ereignis für den anzuschließenden Prozess oder die anzuschließende Aktivität als Eingabeereignis definiert sein.

Wenn Sie mehrere asynchrone Befehle RUN für dieselbe Aktivität innerhalb derselben Arbeitseinheit absetzen, ist Folgendes zu beachten:

- Wenn Sie *dasselbe Eingabeereignis* angeben, schlägt jeder weitere Befehl RUN nach dem ersten fehl.
- Wenn Sie *verschiedene Eingabeereignisse* angeben, kann die Aktivität so oft wie die Anzahl der RUN-Anforderungen aufgerufen werden oder auch nicht. Es ist lediglich garantiert, dass sie mindestens einmal aufgerufen wird. Wenn Sie zum Beispiel fünf asynchrone Anforderungen RUN in derselben Arbeitseinheit für dieselbe Aktivität unter Angabe verschiedener Eingabeereignisse absetzen, könnte die Aktivität zweimal aufgerufen werden. Beim ersten Aufruf könnten drei Eingabeereignisse, beim zweiten zwei Eingabeereignisse übergeben werden.

SYNCHRONOUS

Gibt an, dass der Prozess bzw. die Aktivität synchron zum Requester ausgeführt werden soll.

Bedingungen

107 ACTIVITYBUSY

RESP2-Werte:

19

Die Anforderung hat das Zeitlimit überschritten. Es kann sein, dass die Beendigung einer anderen Task, die diesen Aktivitätsdatensatz verwendet, verhindert wurde.

109 ACTIVITYERR

RESP2-Werte:

8

Die in der Option ACTIVITY angegebene Aktivität konnte nicht gefunden werden.

14

Die auszuführende Aktivität befindet sich nicht im Modus INITIAL oder DORMANT.

27

Die im Befehl RUN SYNCHRONOUS angegebene Aktivität wurde abnormal beendet.

111 EVENTERR

RESP2-Werte:

7

Das in der Option INPUTEVENT angegebene Ereignis wurde von der auszuführenden Aktivität oder dem auszuführenden Prozess nicht als Eingabeereignis definiert oder der Ausführungsstatus des Ereignisses ist FIRED (ausgelöst).

16 INVREQ

RESP2-Werte:

4

Mit der Option ACTIVITY wurde eine untergeordnete Aktivität angegeben, aber der Befehl wurde außerhalb des Geltungsbereichs einer gegenwärtig aktiven Aktivität ausgegeben.

15

Die Task, die den Befehl RUN ACQPROCESS abgesetzt hat, hat keinen Prozess definiert oder abgerufen.

20

Die Option SYNCHRONOUS wurde verwendet, jedoch ist die auszuführende Aktivität ausgesetzt.

24

Die Option ACQACTIVITY wurde verwendet, jedoch hat die Arbeitseinheit, die die Anforderung abgesetzt hat, keine Aktivität angefordert.

28

CICS konnte die Transaktion, die dem auszuführenden Prozess bzw. der auszuführenden Aktivität zugeordnet ist, nicht anschließen. (Diese Antwort tritt nur bei Befehlen RUN SYNCHRONOUS auf.)

32

Die Option SYNCHRONOUS wurde verwendet, jedoch ist die Transaktion, die dem auszuführenden Prozess bzw. der auszuführenden Aktivität zugeordnet ist, als fern definiert. Sie können keinen Prozess und keine Aktivität synchron ausführen, wenn die zugehörige Transaktion fern ist.

40

Das Programm, das den auszuführenden Prozess oder die auszuführende Aktivität implementiert, ist ein fernes Programm.

17 IOERR

RESP2-Werte:

29

Die Repository-Datei ist nicht verfügbar.

30

Es ist ein Ein-/Ausgabefehler für die Repository-Datei aufgetreten.

100 LOCKED

Die Anforderung kann nicht ausgeführt werden, weil eine Sperre für den relevanten Datensatz in der Repository-Datei besteht.

70 NOTAUTH

RESP2-Werte:

101

Der Benutzer, der der absetzenden Task zugeordnet ist, ist nicht berechtigt, den Prozess oder die Aktivität auszuführen.

106 PROCESSBUSY

RESP2-Werte:

13

Die Anforderung hat das Zeitlimit überschritten. Es kann sein, dass die Beendigung einer anderen Task, die diesen Prozessdatensatz verwendet, verhindert wurde.

108 PROCESSERR

RESP2-Werte:

6

Sie können den aktuellen Prozess nicht ausführen.

9

Der Prozesstyp konnte nicht gefunden werden.

14

Der auszuführende Prozess befindet sich nicht im Modus INITIAL oder DORMANT.

27

Der im Befehl RUN SYNCHRONOUS angegebene Prozess wurde abnormal beendet.

RUN TRANSID

Der Befehl RUN TRANSID initialisiert eine untergeordnete Transaktion, die asynchron zur übergeordneten Transaktion ausgeführt wird.

RUN TRANSID

➡ RUN — TRANSID(*name*) — CHANNEL(*name*) — CHILD(*datenbereich*) — ➡

Bedingungen: CHANNELERR, INVREQ, TRANSIDERR, NOTAUTH, DISABLED

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl RUN TRANSID startet eine Task auf dem lokalen System, optional unter Übergabe eines Kanals. Die gestartete Task (untergeordnete Task) wird asynchron zur startenden Task (übergeordnete Task) und mit demselben Sicherheitskontext wie die übergeordnete Task ausgeführt. Die übergeordnete Task verwendet den Kanal, um Eingabedaten an die untergeordnete Task zu übergeben. Die übergeordnete Task erhält eine Rückantwort, wenn der API-Befehl **FETCH CHILD** oder **FETCH ANY** ausgegeben wird.

Wenn eine untergeordnete Task mit dem EXEC CICS-Befehl **RUN TRANSID** gestartet wird, übernimmt sie den Sicherheitskontext der übergeordneten Task. Der Sicherheitskontext wird bei Ausgabe des Befehls **RUN TRANSID** und noch einmal beim Anhängen der untergeordneten Task überprüft. Falls die untergeordnete Task keine Berechtigung für die Ausführung im bestehenden Kontext besitzt, wird sie mit dem Code AASA abgebrochen.

Wenn die (durch TRANSID angegebene) Transaktion mit der Einstellung DYNAMIC(YES) definiert ist, wird diese Option ignoriert und keine Routing-Entscheidung getroffen.

Optionen

CHANNEL(*name*)

Gibt den 1 bis 16 Zeichen langen Namen eines Kanals an, der für die untergeordnete Task verfügbar gemacht werden soll. Zulässige Zeichen sind A-Z a-z 0-9 \$ @ # ./- _ % & ? ! : | " = ~ , ; < > . Führende und eingebettete Leerzeichen sind nicht zulässig. Falls der angegebene Name weniger als 16 Zeichen umfasst, wird er mit nachgestellten Leerzeichen auf 16 Zeichen aufgefüllt. Wenn der Kanal nicht vorhanden ist, wird er erstellt. Dieser neue Kanal bleibt gültig, bis sich die Verbindungsebene ändert. Informationen zum Geltungsbereich von Kanälen finden Sie unter [Geltungsbereich eines Kanals](#).

Kanalnamen haben immer EBCDIC-Format. Die oben angegebene Gruppe der zulässigen Zeichen für Kanalnamen enthält einige Zeichen, die nicht in allen EBCDIC-Codepages dieselbe Darstellung besitzen. Falls Kanäle zwischen Regionen verlagert werden sollen, empfiehlt es sich daher, bei ihrer Benennung ausschließlich die Zeichen A-Z a-z 0-9 & : = , ; < > . - und _ zu verwenden.

Sie können den Kanalnamen DFHTRANSACTION angeben, um einen Transaktionskanal zu verwenden. Ein Transaktionskanal verliert seine Gültigkeit nicht, wenn sich die Verbindungsebene ändert; der Zugriff auf diesen Kanal ist in der Transaktion immer möglich. Weitere Informationen finden Sie unter [Kanäle und Container](#).

Das Programm, das den Befehl **RUN TRANSID** ausgibt, kann Folgendes vornehmen:

- Es kann mit einem oder mehreren Befehlen **PUT CONTAINER CHANNEL** bzw. **PUT64 CONTAINER** den Kanal erstellen, bevor es den Befehl **RUN TRANSID** ausgibt.
- Es kann seinen aktuellen Kanal durch den Namen angeben.
- Es kann einen Kanal benennen, der zurzeit nicht vorhanden ist. In diesem Fall wird ein neuer leerer Kanal erstellt.

Die untergeordnete Task erhält eine *Kopie* der Container des Kanals (und der in ihnen enthaltenen Daten). Die Kopie wird bei der Ausgabe des Befehls **RUN TRANSID** erstellt.

CHILD(datenbereich)

Gibt einen 16 Zeichen umfassenden Binärdatenbereich an, in den CICS das untergeordnete Token stellen soll, das die untergeordnete Task darstellt. Mithilfe des untergeordneten Tokens wird die untergeordnete Task referenziert, damit über einen API-Befehl wie beispielsweise **FETCH CHILD** ein Rückantwort von ihr erhalten werden kann.

TRANSID(name)

Gibt die symbolische ID (1 bis 4 Zeichen) der lokalen Transaktion an, die infolge eines Befehls **RUN** gestartet werden soll.

Bedingungen

122 CHANNELERR

RESP2-Werte:

1

Der in der Option **CHANNEL** angegebene Kanal enthält ein falsches Zeichen oder eine falsche Zeichenkombination.

16 INVREQ

RESP2-Werte:

17

Die gestartete Transaktion ist nicht für einen Systemabschluss geeignet und für die CICS-Region wird gerade ein Systemabschluss durchgeführt. Die Bedingung **INVREQ** (RESP2 nicht festgelegt) tritt ebenfalls auf, falls der Befehl **RUN** nicht für die Verarbeitung durch CICS gültig ist.

19

Der Befehl **RUN TRANSID** wurde während der Transaktionsbeendigung ausgegeben.

28 TRANSIDERR

RESP2-Werte:

1

Die in einem Befehl **RUN TRANSID** angegebene Transaktions-ID ist nicht für CICS definiert.

11

Die angegebene Transaktion ist als ferne Transaktion definiert.

70 NOTAUTH

RESP2-Werte:

101

Der Benutzer, der der ausgebenden Task zugeordnet ist, ist nicht zur Ausführung der untergeordneten Task berechtigt.

84 DISABLED

RESP2-Werte:

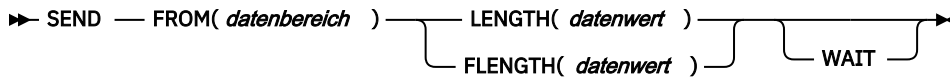
50

Die angegebene Transaktion ist inaktiviert.

SEND (Standardformat für z/OS Communications Server)

Schreibt Daten in die standardmäßige CICS-Terminalunterstützung.

SEND (Standardformat für z/OS Communications Server)



Bedingungen: INVREQ, LENGERR, NOTALLOC

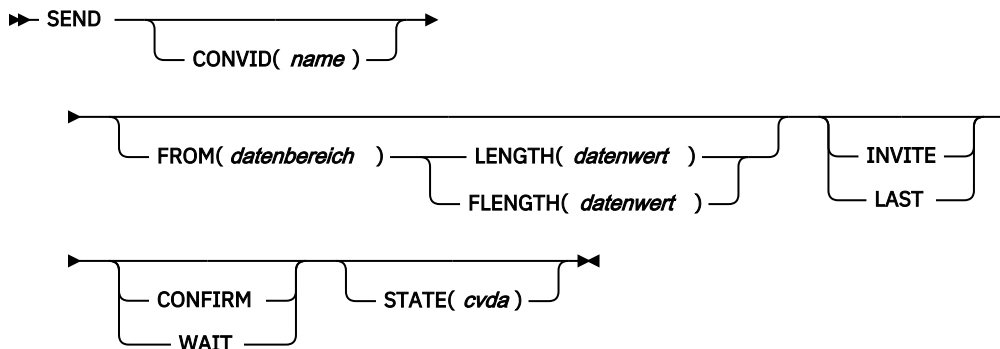
Beschreibung

Der Befehl SEND schreibt Daten in ein Terminal. Diese Form des Befehls SEND kann von allen durch CICS unterstützten Terminals verwendet werden, für die die anderen SEND-Beschreibungen nicht geeignet sind.

SEND (APPC)

Sendet Daten bei einem formatfreien APPC-Datenaustausch.

SEND (APPC)



Bedingungen: INVREQ, LENGERR, NOTALLOC, SIGNAL, TERMERR

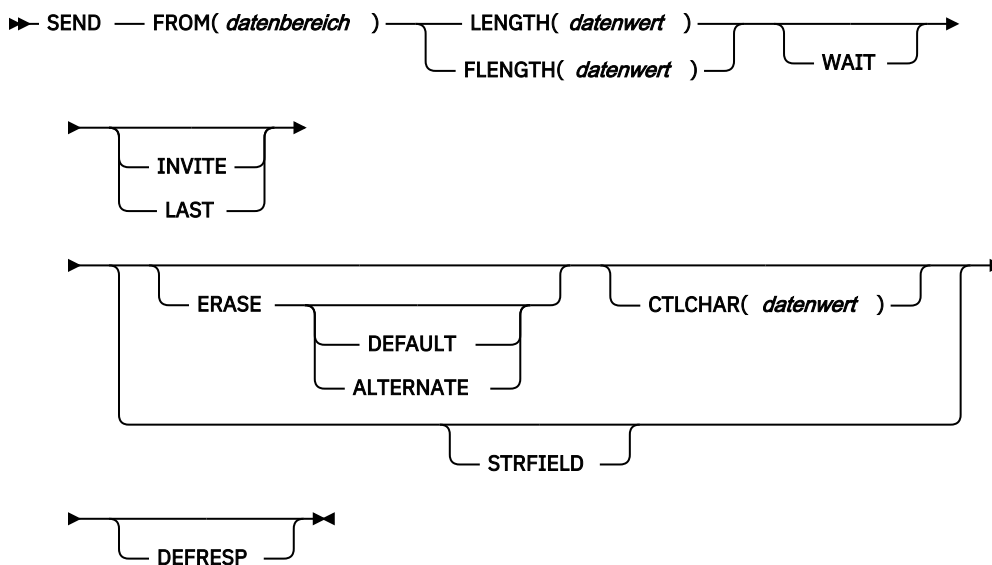
Beschreibung

Der Befehl SEND sendet Daten bei einem formatfreien APPC-Datenaustausch an den Dialogpartner.

SEND (LUTYPE2/LUTYPE3)

Schreibt Daten in eine logische Einheit mit 3270-Anzeige (LUTYPE2) oder mit 3270-Drucker (LUTYPE3).

SEND (LUTYPE2/LUTYPE3)



Bedingungen: INVREQ, LENGERR, TERMERR

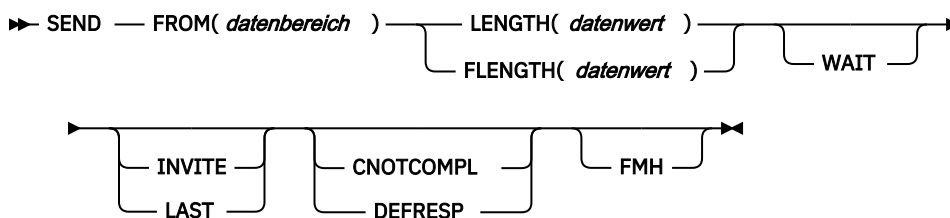
Beschreibung

Der Befehl **SEND** schreibt Daten in ein Terminal.

SEND (LUTYPE4)

Schreibt Daten in eine logische Einheit mit LU-Typ 4.

SEND (LUTYPE4)



Bedingungen: INVREQ, IREQCD, LENGERR, SIGNAL, TERMERR

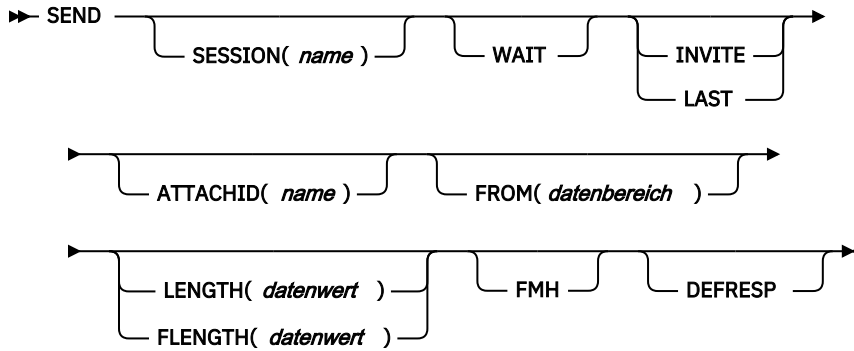
Beschreibung

Der Befehl SEND schreibt Daten in ein Terminal.

SEND (LUTYPE6.1)

Sendet Daten bei einem Datenaustausch mit LU-Typ 6.1.

SEND (LUTYPE6.1)



Bedingungen: CBIDERR, INVREQ, LENGERR, NOTALLOC, SIGNAL, TERMERR

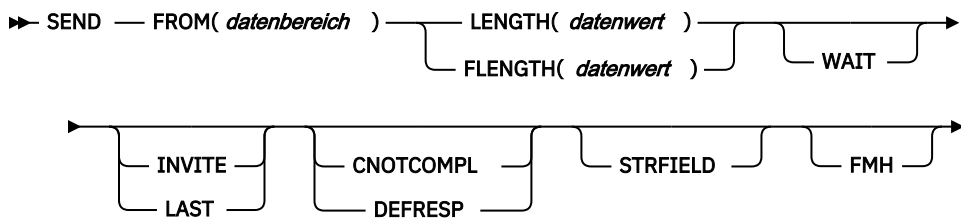
Beschreibung

Der Befehl SEND sendet Daten bei einem Datenaustausch mit LU-Typ 6.1 an den Dialogpartner.

SEND (SCS)

Schreibt Daten in eine logische Einheit mit 3270-SCS-Drucker.

SEND (SCS)



Bedingungen: INVREQ, LENGERR, TERMERR

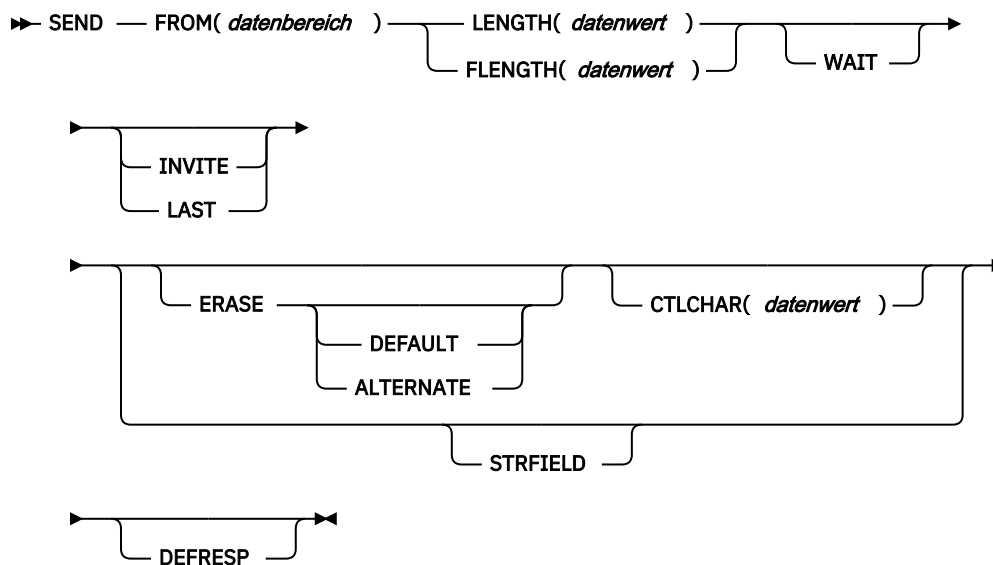
Beschreibung

Der Befehl SEND schreibt Daten in eine logische Einheit. Die logische Einheit mit SCS-Drucker akzeptiert eine Zeichenfolge gemäß der Definition durch Systems Network Architecture (SNA).

SEND (3270 logisch)

Schreibt Daten in eine logische 3270-Einheit.

SEND (3270 logisch)



Bedingungen: INVREQ, LENGERR, TERMERR

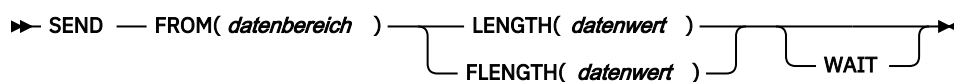
Beschreibung

Der Befehl SEND schreibt Daten in ein Terminal.

SEND (3600-Pipeline)

Schreibt Daten in eine logische 3600-Pipeline-Einheit.

SEND (3600-Pipeline)



Bedingungen: INVREQ, LENGERR, TERMERR

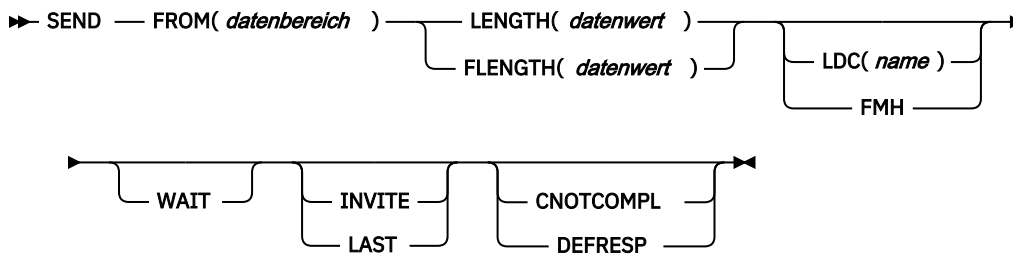
Beschreibung

Der Befehl SEND schreibt Daten in ein Terminal.

SEND (3600-3601)

Schreibt Daten in eine logische 3600- bzw. 3601-Einheit.

SEND (3600-3601)



Bedingungen: INVREQ, LENGERR, SIGNAL, TERMERR

Beschreibung

Der Befehl SEND schreibt Daten in ein Terminal. Dieses Format des Befehls SEND gilt auch für das System IBM 4770 und das Werkskommunikationssystem IBM 3630 (Plant Communication System).

Ein logischer Einheitencode (Logical Device Code, LDC) ist ein Code, der in einen abgehenden Funktionsverwaltungsheader aufgenommen werden kann, um die Disposition der Daten anzugeben (beispielsweise das Subsystemterminal, an das die Daten gesendet werden sollen). Jeder Code kann durch ein eindeutiges mnemonisches LDC-Zeichen dargestellt werden.

Die Installation kann bis zu 256 zweistellige mnemonische Zeichen für jeden Eintrag der Terminalsteuertabelle (TCTTE) angeben; zwei oder mehr Einträge der Terminalsteuertabelle können eine Liste dieser mnemonischen Zeichen gemeinsam nutzen. Jedem mnemonischen LDC-Zeichen für jeden Eintrag der Terminalsteuertabelle entspricht ein numerischer Wert (0 bis 255).

Einem logischen Einheitencode sind außerdem eine 3600-Einheit sowie eine logische Seitengröße zugeordnet. *LDC* oder *LDC-Wert* bezieht sich in den vorliegenden Informationen auf den Code, der vom Benutzer angegeben wird. Mit dem Begriff *mnemonisches LDC-Zeichen* ist das zweistellige Symbol gemeint, das den numerischen LDC-Wert darstellt.

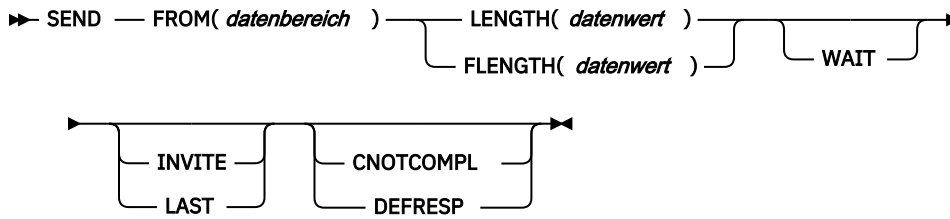
Wenn die Option LDC angegeben ist, wird in den Funktionsverwaltungsheader (FMH) der numerische Wert eingefügt, der dem mnemonischen Zeichen des jeweiligen Eintrags der Terminalsteuertabelle zugeordnet ist. Der numerische Wert, der dem mnemonischen LDC-Zeichen zugeordnet ist, wird von der Installation ausgewählt und durch das 3601-Anwendungsprogramm interpretiert.

Bei der Ausgabe kann der Funktionsverwaltungsheader durch das Anwendungsprogramm oder durch CICS erstellt werden. Falls Ihr Programm den Funktionsverwaltungsheader angibt, platzieren Sie ihn am Beginn Ihrer Ausgabedaten und geben Sie die Option FMH im Befehl SEND an. Wenn Sie die Option FMH weglassen, stellt CICS einen Funktionsverwaltungsheader bereit; Sie müssen allerdings die ersten drei Byte der Nachricht zur Vervollständigung durch CICS reservieren.

SEND (3600-3614)

Schreibt Daten in eine logische 3600- bzw. 3614-Einheit.

SEND (3600-3614)



Bedingungen: INVREQ, LENGERR, TERMERR

Beschreibung

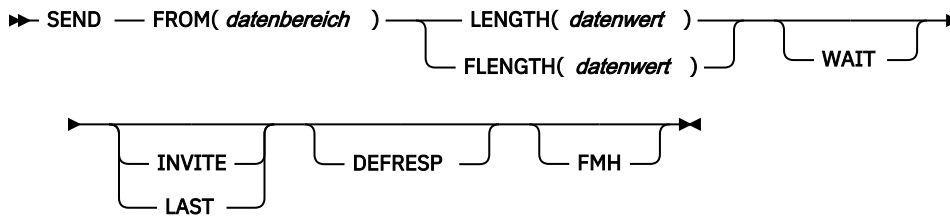
Der Befehl SEND schreibt Daten in ein Terminal. Das Datenstrom- und Kommunikationsformat, das zwischen einem CICS-Anwendungsprogramm und einer 3614-Einheit verwendet wird, wird durch die 3614-Einheit bestimmt. Das Anwendungsprogramm ist somit bei der Verarbeitung der 3614-Kommunikation einheitenabhängig.

Weitere Informationen zum Entwerfen von 3614-Anwendungsprogrammen für CICS enthält das Handbuch zum System *IBM 4700/3600/3630*.

SEND (3650-Interpreter)

Schreibt Daten in eine logische Einheit mit 3650-Interpreter.

SEND (3650-Interpreter)



Bedingungen: INVREQ, LENGERR, TERMERR

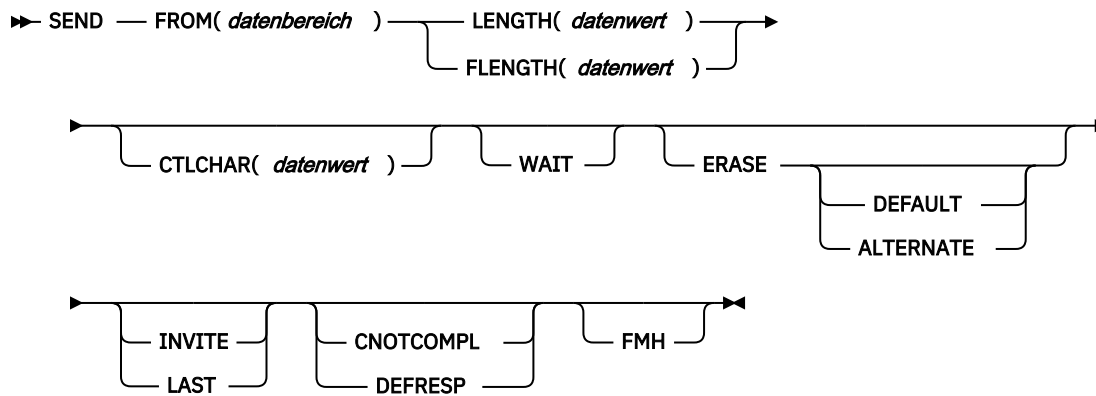
Beschreibung

Der Befehl SEND schreibt Daten in ein Terminal.

SEND (3650-3270)

Schreibt Daten in eine logische 3650-Einheit.

SEND (3650-3270)



Bedingungen: INVREQ, LENGERR, TERMERR

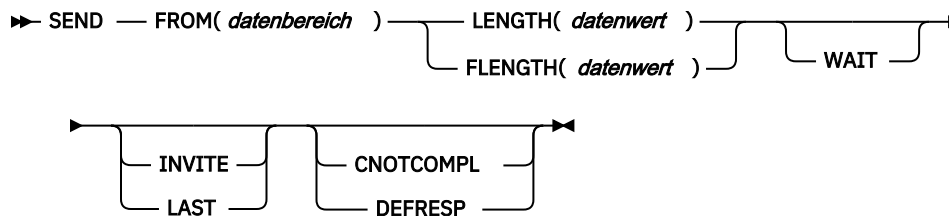
Beschreibung

Der Befehl SEND schreibt Daten in ein Terminal.

SEND (3650-3653)

Schreibt Daten in eine logische 3650- bzw. 3653-Einheit.

SEND (3650-3653)



Bedingungen: INVREQ, LENGERR, TERMERR

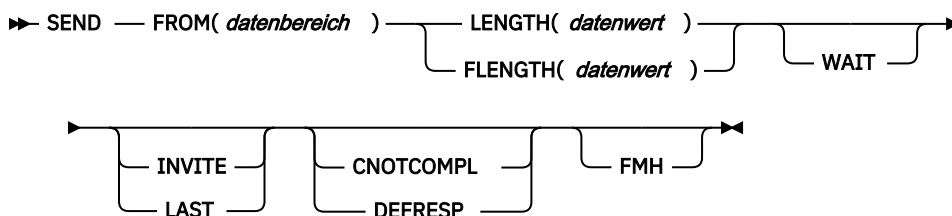
Beschreibung

Der Befehl SEND schreibt Daten in ein Terminal.

SEND (logische 3650- bzw. 3680-Einheit)

Schreibt Daten in eine logische 3650- bzw. 3680-Einheit.

SEND (3650-3680)



Bedingungen: INVREQ, LENGERR, TERMERR

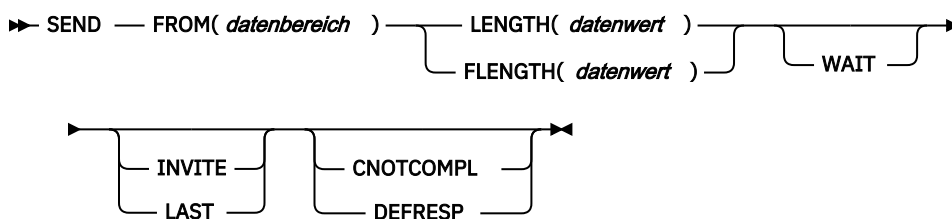
Beschreibung

Der Befehl SEND schreibt Daten in ein Terminal.

SEND (3767)

Schreibt Daten in eine interaktive logische 3767-Einheit.

SEND (3767)



Bedingungen: INVREQ, LENGERR, SIGNAL, TERMERR

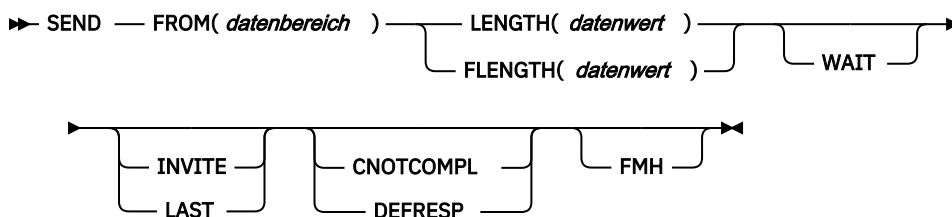
Beschreibung

Der Befehl SEND schreibt Daten in ein Terminal. Dieses Format des Befehls SEND gilt auch für die interaktive logische 3770-Einheit.

SEND (3770)

Schreibt Daten in eine logische Einheit mit 3770-Stapel.

SEND (3770)



Bedingungen: INVREQ, LENGERR, SIGNAL, TERMERR

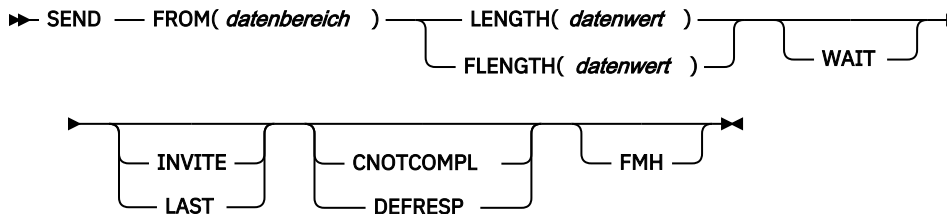
Beschreibung

Der Befehl SEND schreibt Daten in ein Terminal.

SEND (3790 mit vollem Funktionsumfang oder Anfrage)

Schreibt Daten in eine logische 3790-Einheit mit vollem Funktionsumfang oder Anfrage).

SEND (3790 mit vollem Funktionsumfang oder Anfrage)



Bedingungen: INVREQ, LENGERR, SIGNAL, TERMERR

Beschreibung

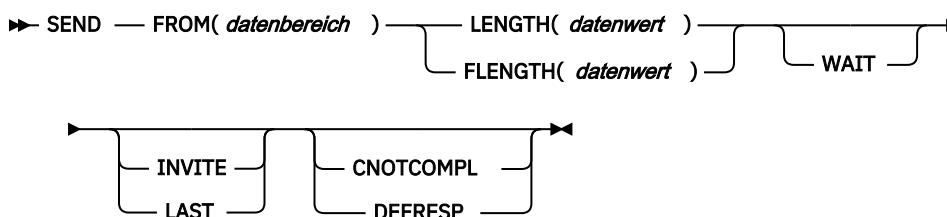
Der Befehl SEND schreibt Daten in ein Terminal. Dieses Format des Befehls SEND gilt auch für die folgenden Einheiten:

- Logische 3650- bzw. 3680-Einheit mit vollem Funktionsumfang
- Logische 3770-Einheit mit vollem Funktionsumfang

SEND (3790 SCS)

Schreibt Daten in eine logische Einheit mit 3790-SCS-Drucker.

SEND (3790 SCS)



Bedingungen: INVREQ, LENGERR, TERMERR

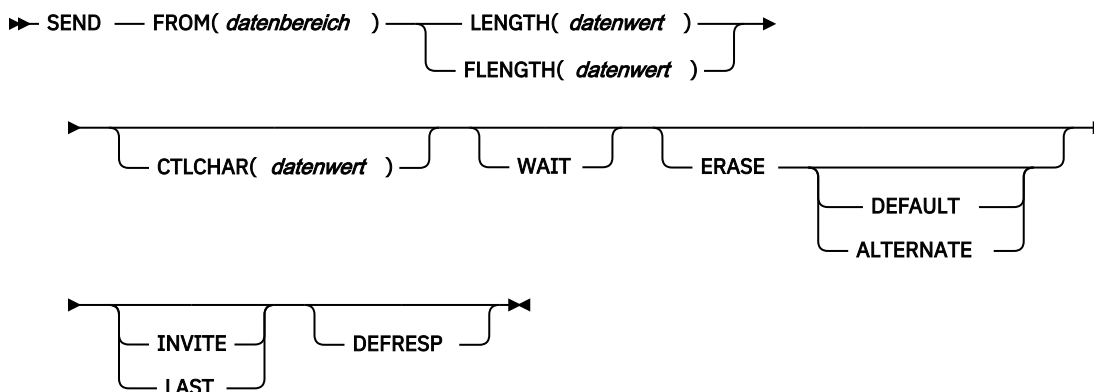
Beschreibung

Der Befehl SEND schreibt Daten in ein Terminal.

SEND (3790- bzw. 3270-Anzeige)

Schreibt Daten in eine logische Einheit mit 3790- bzw. 3270-Anzeige.

SEND (3790- bzw. 3270-Anzeige)



Bedingungen: INVREQ, LENGERR, TERMERR

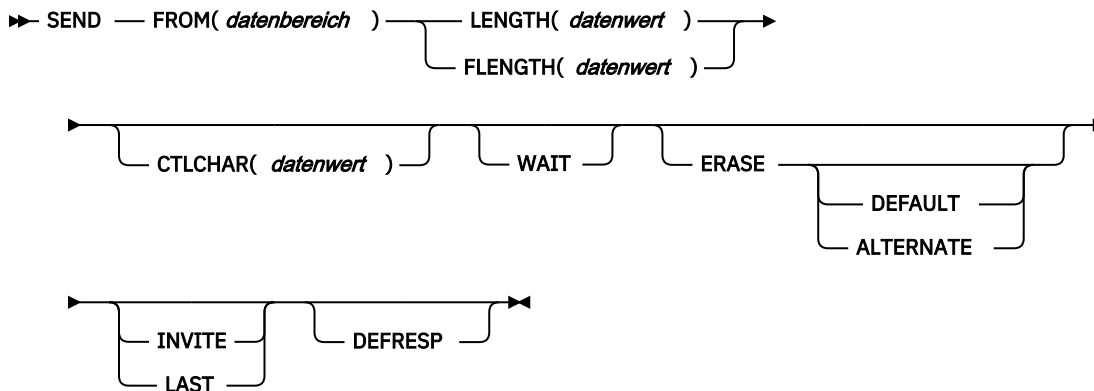
Beschreibung

Der Befehl SEND schreibt Daten in ein Terminal.

SEND (3790- bzw. 3270-Drucker)

Schreibt Daten in eine logische Einheit mit 3790- bzw. 3270-Drucker.

SEND (3790- bzw. 3270-Drucker)



Bedingungen: INVREQ, LENGERR, TERMERR

Beschreibung

Der Befehl SEND schreibt Daten in ein Terminal.

SEND: Optionen für z/OS Communications Server

In diesem Abschnitt sind allgemeine Optionen beschrieben, die für den Befehl SEND (bei z/OS Communications Server) verwendet werden.

Optionen

ALTERNATE

Legt die Verwendung der alternativen Anzeigegröße durch das Terminal fest.

ATTACHID(name)

Gibt an, dass ein Anschlussheader (erstellt durch einen Befehl **BUILD ATTACH**) den Benutzerdaten, die in der Option FROM bereitgestellt werden, vorangehen und mit diesen verkettet werden soll. Der Wert "name" (1-8 Zeichen) identifiziert den Anschlussheadersteuerblock, der in der lokalen Task verwendet werden soll.

CNOTCOMPL

Gibt an, dass die Anforderung/Antwort-Einheit, die als Ergebnis dieses Befehls **SEND** gesendet wird, die Kette nicht abschließt. Falls diese Option weggelassen wird und die Kettenassemblierung angegeben wurde, beendet die Anforderung/Antwort-Einheit die Kette.

CONFIRM

Gibt an, dass die Anwendung, die einen Datenaustausch mit der Synchronisationsebene 1 oder 2 verwendet, eine Antwort von der fernen Anwendung erforderlich macht. Eine ferne CICS-Anwendung kann durch die Ausführung eines Befehls **ISSUE CONFIRMATION** eine positive Antwort geben bzw. durch die Ausführung eines Befehls **ISSUE ERROR** negativ antworten; in diesem Fall wird für die sendende Anwendung EIBERR und EIBERRCD festgelegt. CICS gibt die Steuerung erst dann an die sendende Anwendung zurück, nachdem die Antwort empfangen wurde.

CONVID(name)

Gibt den Dialog an, auf den sich der Befehl bezieht. Der vier Zeichen lange Name gibt entweder das durch einen vorherigen Befehl **ALLOCATE** im Feld EIBRSRCE des EIB zurückgegebene Token oder aber das Token an, das die Hauptsitzung darstellt (und durch einen vorherigen Befehl **ASSIGN** zurückgegeben wurde).

Aus Gründen der Kompatibilität mit früheren Releases wird SESSION als Synonym für CONVID akzeptiert. Neue Programme müssen CONVID verwenden.

Falls diese Option nicht angegeben wird, wird die Hauptfunktion verwendet.

CTLCHAR(datenwert)

Gibt ein 1 Byte langes Schreibbefehlssteuerzeichen an, das einen Befehl **SEND** für eine 3270-Einheit steuert. Diese Steuerzeichen sind in der Veröffentlichung [IBM 3270 Data Stream Programmers Reference](#) beschrieben. Ein COBOL-Benutzer muss einen Datenbereich angeben, der dieses Zeichen enthält. Wenn die Option nicht angegeben wird, werden alle Tags für modifizierte Daten (MDTs) auf null zurückgesetzt und die Tastatur wird wiederhergestellt.

DEFAULT

Legt die Verwendung der Standardanzeigegröße durch das Terminal fest.

DEFRESP

Gibt an, dass nach Abschluss der Ausgabeoperation eine unbedingte Antwort erforderlich ist.

ERASE

Gibt an, dass der Anzeigendruckerpuffer oder die Partition gelöscht und der Cursor in die linke obere Ecke der Anzeige zurückgesetzt werden soll. Diese Option gilt nur für die 3270- bzw. 8775-Anzeige und die 3604-Tastaturanzeige.

Die erste Ausgabeoperation in einer Transaktion oder in einer Serie von Pseudodialogtransaktionen muss immer die Option ERASE angeben. Für Transaktionen, die an 3270-Anzeigen oder -Drucker angehängt sind, stellt diese Option (sofern sie nicht explizit durch die Option DEFAULT oder ALTERNATE überschrieben wird) außerdem sicher, dass die richtige Anzeigegröße ausgewählt wird, die für die Transaktion durch die Option SCRNSIZE in der Ressourcendefinition PROFILE definiert ist.

FLENGTH(datenwert)

Dies ist eine Alternative zur Option LENGTH. Aus architekturabhängigen Gründen ist diese Option bei allen terminalbezogenen Befehlen **SEND** auf 32 KB begrenzt.

FMH

Gibt an, dass in die zu schreibenden Daten des Bereichs ein Funktionsverwaltungsheader eingeschlossen ist. Wenn ebenfalls die Option ATTACHID angegeben ist, wird die Markierung für verkettete Funktionsverwaltungsheader im Funktionsverwaltungsheader für den Anschluss festgelegt.

Die Verwendung von FMH ist optional und wird nicht von allen Terminaltypen unterstützt. Wenn kein Funktionsverwaltungsheader angegeben ist, führt CICS keine Aktion aus, außer bei 3600-/4700-Terminals; hier ist ein Funktionsverwaltungsheader obligatorisch. Wenn in einem solchen Fall kein Funktionsverwaltungsheader angegeben ist, stellt CICS einen Funktionsverwaltungsheader bereit und platziert ihn in den ersten drei Byte der Nachricht, die Sie für diesen Zweck reservieren müssen.

FROM(datenbereich)

Gibt die Daten an, die in die logische Einheit oder in eine Partnertransaktion geschrieben werden sollen.

INVITE

Beim Befehl **SEND** (APPC) kann eine Anwendung mit der Option INVITE einen Anzeiger für die Änderung der Übertragungsrichtung zu Daten hinzufügen, die bereits an einen Prozess in einem verbundenen APPC-System gesendet wurden. Steuerdaten werden durch CICS erst bei der nachfolgenden Option WAIT oder einem Befehl **SYNCPPOINT** übertragen, sofern nicht die Option CONFIRM oder WAIT ebenfalls im Befehl **GDS SEND INVITE** codiert ist.

Bei den anderen Befehlen **SEND** gibt die Option INVITE an, dass der nächste Terminalsteuerungsbefehl, der für diese Funktion ausgeführt werden soll, ein Befehl RECEIVE ist. Dies ermöglicht optimale Datenflüsse.

LAST

Gibt an, dass es sich bei diesem Befehl um den letzten Befehl **SEND** für eine Transaktion handelt.

LDC(name)

Gibt das zweistellige mnemonische Zeichen an, mit dem der numerische Wert für den entsprechenden logischen Einheitencode ermittelt wird. Das mnemonische Zeichen stellt einen Eintrag für den logischen Einheitencode (Logical Device Code, LDC) in der Terminalsteuertabelle TYPE=LDC dar.

LENGTH(datenwert)

Gibt als Halbwort-Binärwert die Länge der zu schreibenden Daten an. Eine Beschreibung einer sicheren Obergrenze finden Sie unter [„LENGTH-Optionen in CICS-Befehlen“](#) auf Seite 10.

SESSION(name)

Gibt die symbolische ID (1-4 Zeichen) für einen Terminaleintrag in der Terminalsteuertabelle (TCTTE) für die Sitzung an. Diese Option gibt die zu verwendende alternative Funktion an. Falls sie nicht angegeben ist, wird die Hauptfunktion für die Task verwendet.

STATE(cvda)

Ruft den Status des aktuellen Datenaustausches ab. Von CICS werden die folgenden CVDA-Werte zurückgegeben:

- ALLOCATED
- CONFFREE
- CONFRECEIVE
- CONFSEND
- FREE
- PENDFREE
- PENDRECEIVE
- RECEIVE
- ROLLBACK
- SEND
- SYNCFREE
- SYNCRECEIVE

- SYNCSEND

STRFIELD

Gibt an, dass der Datenbereich, der in der Option FROM angegeben ist, strukturierte Felder enthält. Wenn diese Option angegeben wird, muss der Inhalt aller strukturierten Felder vom Anwendungsprogramm verarbeitet werden. Anstelle des Befehls **SEND** muss der Befehl **CONVERSE** verwendet werden, falls der Datenbereich ein strukturiertes Feld für eine Lesepartition enthält. Strukturierte Felder sind in der Veröffentlichung IBM 3270 Data Stream Programmers Reference beschrieben.

Die Optionen CTLCHAR bzw. ERASE und STRFIELD schließen sich gegenseitig aus; ihre Verwendung in Kombination mit STRFIELD generiert eine Fehlermeldung.

WAIT

Gibt an, dass die Verarbeitung des Befehls abgeschlossen sein muss, bevor eine nachfolgende Verarbeitung versucht wird.

Falls die Option WAIT nicht angegeben ist, wird die Steuerung an das Anwendungsprogramm zurückgegeben, sobald die Verarbeitung des Befehls gestartet wird. Eine nachfolgende Eingabe- oder Ausgabeanforderung (Terminalsteuerung, BMS oder Stapeldatenaustausch) an das Terminal, das der Task zugeordnet ist, führt dazu, dass sich das Anwendungsprogramm im Wartestatus befindet, bis die vorherige Anforderung abgeschlossen wurde.

Bedingungen

Einige der folgenden Bedingungen können kombiniert auftreten. Falls mehrere Bedingungen auftreten, wird nur die erste Bedingung an das Anwendungsprogramm übergeben. EIBRCODE wird jedoch so festgelegt, dass alle aufgetretenen Bedingungen angegeben sind.

62 CBIDERR

Tritt auf, falls der angeforderte Steuerblock des Anschlussheaders, der in ATTACHID angegeben ist, nicht gefunden wurde.

Standardaktion: Die Task wird abnormal beendet.

57 IGREQCD

Tritt auf, wenn versucht wird, einen Befehl **SEND** auszuführen, nachdem ein Datenflusssteuerungsbeefehl SIGNAL mit einem Code für die Anforderung der Richtungsänderung (Request change direction, RCD) von der logischen Einheit empfangen wurde.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

RESP2-Werte:

200

Eine Serveranwendung für Distributed Program Link hat versucht, Daten in ihrer Sitzung für die Funktionsverlagerung (ihrer Hauptfunktion) zu senden.

Beim Befehl **SEND** (APPC) hat eine Serveranwendung für Distributed Program Link in der Option CONVID die Sitzung für die Funktionsverlagerung (ihre Hauptfunktion) angegeben.

Tritt außerdem in einer der folgenden Situationen auf (RESP2 nicht festgelegt):

- Die Option CONFIRM war angegeben, aber der APPC-Datenaustausch hat nicht die Synchronisationsebene 1 oder 2.
- Der Befehl SEND wurde für einen APPC-Datenaustausch verwendet, der kein formatfreier Datenaustausch ist oder nicht die EXEC-Schnittstelle von CICS nutzt.

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

Tritt auf, falls in der Option LENGTH oder FLENGTH ein Wert angegeben ist, der außerhalb des gültigen Bereichs liegt.

Standardaktion: Die Task wird abnormal beendet.

61 NOTALLOC

Tritt auf, falls sich der Wert der Option CONVID im Befehl nicht auf einen Datenaustausch bezieht, dessen Eigner die Anwendung ist, oder falls die Anwendung nicht Eigner der im Befehl angegebenen Funktion ist.

Standardaktion: Die Task wird abnormal beendet.

24 SIGNAL

Tritt auf, wenn ein eingehender Befehl SIGNAL für die Datenflusssteuerung von einer logischen Einheit oder Sitzung empfangen wird. Beim Empfang eines eingehenden Signals wird das Feld EIBSIG immer festgelegt.

Standardaktion: Die Bedingung wird ignoriert.

81 TERMERR

Tritt für einen sitzungsbezogenen Fehler auf. Jede Aktion außer einem Befehl FREE für diesen Dialog verursacht einen Abbruch mit dem Code ATCV.

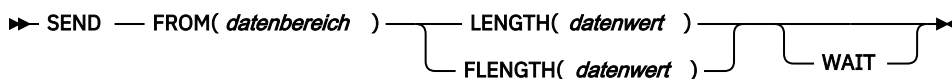
Diese Bedingung kann durch eine Anforderung CANCEL TASK durch ein Knotenfehlerprogramm des Benutzers verursacht werden, falls für die Task eine Terminalsteuerungsanforderung aussteht, wenn der Sitzungsfehler durch das Programm des Knotens für abnormale Bedingungen abgewickelt wird.

Standardaktion: Die Task wird abnormal mit dem Abbruchcode ATNI beendet.

SEND (Standardformat ohne z/OS Communications Server)

Schreibt Daten an die standardmäßige CICS-Terminalunterstützung.

SEND (Standardformat ohne z/OS Communications Server)



Bedingungen: INVREQ, LENGERR, NOTALLOC

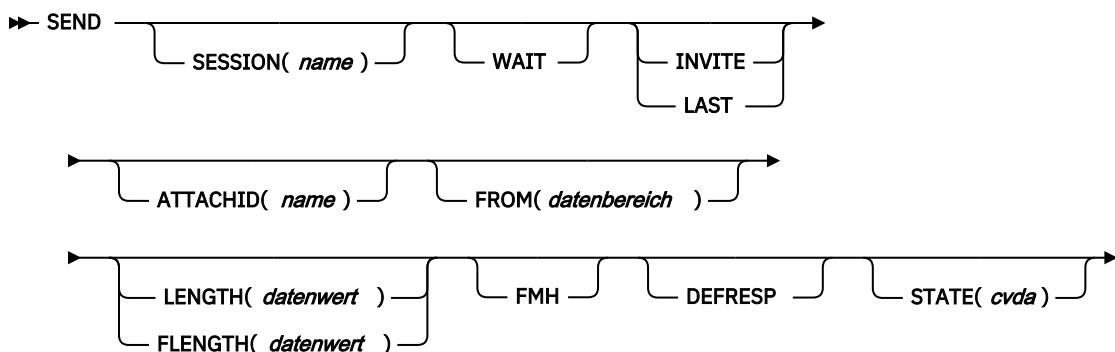
Beschreibung

Der Befehl SEND schreibt Daten in ein Terminal. Diese Form des Befehls SEND kann von allen durch CICS unterstützten Terminals verwendet werden, für die die anderen SEND-Beschreibungen nicht geeignet sind.

SEND (MRO)

Sendet Daten in einen MRO-Dialog.

SEND (MRO)



Bedingungen: CBIDERR, INVREQ, LENGERR, NOTALLOC, TERMERR

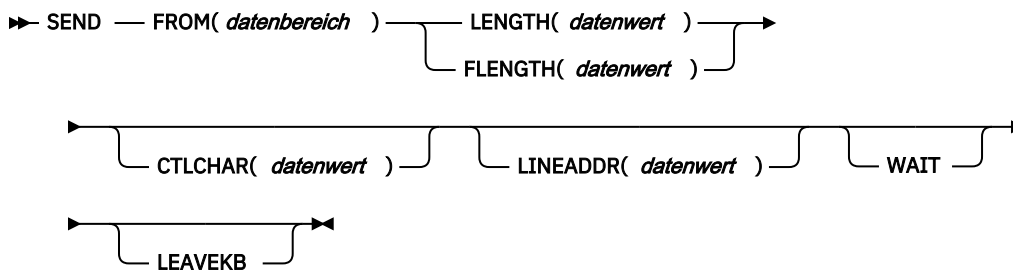
Beschreibung

Der Befehl SEND sendet Daten bei einem MRO-Dialog an den Dialogpartner.

SEND (2260)

Schreibt Daten in eine 2260- oder 2265-Anzeigestation.

SEND (2260)



Bedingungen: INVREQ, LENGERR

Beschreibung

Der Befehl SEND schreibt Daten in ein Terminal.

SEND (2980)

Schreibt Daten in ein Terminalsystem IBM 2980 für allgemeine Bankgeschäfte.

SEND (2980)



Bedingungen: INVREQ, LENGERR, NOPASSBKWR

Beschreibung

Der Befehl SEND schreibt Daten in ein Terminal. Weitere Informationen zum System IBM 2980 für allgemeine Bankgeschäfte finden Sie unter [„RECEIVE \(2980\)“](#) auf Seite 426.

SEND: Optionen für Nicht-z/OS Communications Server

In diesem Abschnitt sind allgemeine Optionen beschrieben, die für den Befehl SEND verwendet werden (nicht bei z/OS Communications Server).

Optionen

ALTERNATE

Legt die Verwendung der alternativen Anzeigegröße durch das Terminal fest.

ASIS

Gibt an, dass die Ausgabe im Transparenzmodus gesendet werden soll, also ohne Erkennung von Steuerzeichen und unter Akzeptanz aller 256 möglichen 8-Bit-Kombinationen als gültige und übertragbare Daten.

Anmerkung: Falls Sie ein Katakana-Terminal verwenden, werden möglicherweise einige Nachrichten angezeigt, in denen englische und Katakana-Zeichen kombiniert sind. Dies liegt daran, dass Katakana-Terminals keine Ausgabe in Groß-/Kleinschreibung anzeigen können. Großbuchstaben im Datenstrom werden als englische Großbuchstaben angezeigt, Kleinbuchstaben jedoch als Katakana-Zeichen. Bitten Sie in einem solchen Fall Ihren Systemprogrammierer, in den Systeminitialisierungsparametern MSGCASE=UPPER anzugeben, damit diese Nachrichten ausschließlich Großbuchstaben enthalten.

ATTACHID(name)

Gibt an, dass ein Anschlussheader (erstellt durch einen Befehl BUILD ATTACH) den Benutzerdaten, die in der Option FROM bereitgestellt werden, vorangehen und mit diesen verkettet werden soll. Der Wert "name" (1-8 Zeichen) identifiziert den Anschlussheadersteuerblock, der in der lokalen Task verwendet werden soll.

CBUFF

Gibt an, dass die Daten in einem 2972-Controller in einen allgemeinen Puffer geschrieben werden sollen. Die Option WAIT wird impliziert.

CNOTCOMPL

Gibt an, dass die Anforderung/Antwort-Einheit, die als Ergebnis dieses Befehls SEND gesendet wird, die Kette nicht abschließt. Falls diese Option weggelassen wird und die Kettenassemblierung angegeben wurde, beendet die Anforderung/Antwort-Einheit die Kette.

CTLCHAR(datenwert)

Gibt ein 1 Byte langes Schreibbefehlssteuerzeichen an, das einen Befehl SEND für eine 3270-Einheit steuert. Diese Steuerzeichen sind in der Veröffentlichung IBM 3270 Data Stream Programmers Reference dokumentiert. Ein COBOL-Benutzer muss einen Datenbereich angeben, der dieses Zeichen enthält. Wenn die Option nicht angegeben wird, werden alle Tags für modifizierte Daten (MDTs) auf null zurückgesetzt und die Tastatur wird wiederhergestellt.

DEFAULT

Legt die Verwendung der Standardanzeigegröße durch das Terminal fest.

DEFRESP

Gibt an, dass nach Abschluss der Ausgabeoperation eine unbedingte Antwort erforderlich ist.

ERASE

Gibt an, dass der Anzeigendruckerpuffer oder die Partition gelöscht und der Cursor in die linke obere Ecke der Anzeige zurückgesetzt werden soll. (Diese Option gilt nur für die 3270- bzw. 8775-Anzeige und die 3604-Tastaturanzeige.)

Die erste Ausgabeoperation in einer Transaktion oder in einer Serie von Pseudodialogtransaktionen sollte immer die Option ERASE angeben. Für Transaktionen, die an 3270-Anzeigen oder -Drucker angehängt sind, stellt diese Option (sofern sie nicht explizit durch die Option DEFAULT oder ALTERNATE überschrieben wird) außerdem sicher, dass die richtige Anzeigegröße ausgewählt wird, die für die Transaktion durch die Option SCRNSIZE in der RDO-Ressourcendefinition PROFILE definiert ist.

FLENGTH(datenwert)

Dies ist eine Vollwortalternative zu LENGTH.

FMH

Gibt an, dass in die Daten des Bereichs der Option FROM ein Funktionsverwaltungsheader eingeschlossen ist. Wenn ebenfalls die Option ATTACHID angegeben ist, wird die Markierung für verkettete Funktionsverwaltungsheader im Funktionsverwaltungsheader für den Anschluss festgelegt.

FROM(datenbereich)

Gibt die Daten an, die in die logische Einheit oder das Terminal geschrieben werden sollen.

INVITE

Gibt an, dass der nächste Terminalsteuerungsbefehl, der für diese Funktion ausgeführt werden soll, ein Befehl RECEIVE ist. Dies ermöglicht optimale Datenflüsse.

LAST

Gibt an, dass es sich bei dieser Operation um die letzte Ausgabeoperation für eine Transaktion und daher um das Ende einer Klammer handelt.

LEAVEKB

Gibt an, dass die Tastatur nach Abschluss der Datenübertragung gesperrt bleibt.

LENGTH(datenwert)

Gibt als Halbwort-Binärwert die Länge der zu schreibenden Daten an. Eine Beschreibung einer sicheren Obergrenze finden Sie unter „LENGTH-Optionen in CICS-Befehlen“ auf Seite 10.

LINEADDR(datenwert)

Gibt an, dass der Schreibvorgang bei einer bestimmten Zeile einer 2260/2265-Anzeige beginnen soll. Der Datenwert ist ein Halbwort-Binärwert im Bereich von 1 bis 12 (2260-Anzeige) bzw. 1 bis 15 (2265-Anzeige).

PASSBK

Gibt an, dass die Kommunikation mit einem Sparbuch erfolgt. Die Option WAIT wird impliziert.

PSEUDOBIN (nur_start-stop)

Gibt an, dass die geschriebenen Daten aus der hexadezimalen Darstellung von System/7 in die pseudobinäre Darstellung umgesetzt werden sollen.

SESSION(name)

Gibt die symbolische ID (1-4 Zeichen) für einen Terminaleintrag in der Terminalsteuertabelle (TCTTE) für die Sitzung an. Diese Option gibt die zu verwendende alternative Funktion an. Falls sie nicht angegeben ist, wird die Hauptfunktion für die Task verwendet.

STATE(cvda)

Ruft den Status des Transaktionsprogramms ab. Die folgenden CVDA-Werte können von CICS zurückgegeben werden:

- ALLOCATED
- FREE
- PENDFREE
- RECEIVE
- ROLLBACK
- SEND
- SYNCFREE
- SYNCRECEIVE
- SYNCSEND

STRFIELD

Gibt an, dass der Datenbereich, der in der Option FROM angegeben ist, strukturierte Felder enthält. Falls diese Option angegeben ist, muss der Inhalt aller strukturierten Felder durch das Anwendungsprogramm verarbeitet werden. Anstelle des Befehls SEND muss der Befehl CONVERSE verwendet werden, falls der Datenbereich ein strukturiertes Feld für eine Lesepartition enthält. (Strukturierte Felder sind in der Veröffentlichung [IBM 3270 Data Stream Programmers Reference](#) beschrieben.)

Die Optionen CTLCHAR und ERASE schließen die Option STRFIELD aus. Werden sie zusammen mit STRFIELD verwendet, wird eine Fehlermeldung generiert.

WAIT

Gibt an, dass die Verarbeitung des Befehls abgeschlossen sein muss, bevor eine nachfolgende Verarbeitung versucht wird.

Falls die Option WAIT nicht angegeben ist, wird die Steuerung an das Anwendungsprogramm zurückgegeben, sobald der Befehl gestartet worden ist. Eine nachfolgende Eingabe- oder Ausgabeanforderung (Terminalsteuerung, BMS oder Stapeldatenaustausch) an das Terminal, das der Task zugeordnet ist, führt dazu, dass sich das Anwendungsprogramm im Wartestatus befindet, bis die vorherige Anforderung abgeschlossen wurde.

Bedingungen

62 CBIDERR

Tritt auf, falls der angeforderte Steuerblock des Anschlussheaders, der in ATTACHID angegeben ist, nicht gefunden wurde.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

RESP2-Werte:

200

Tritt auf, falls eine Serveranwendung für Distributed Program Link versucht hat, Daten in ihrer Sitzung für die Funktionsverlagerung (ihrer Hauptfunktion) zu senden.

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

Tritt auf, falls in der Option LENGTH oder FLENGTH ein Wert angegeben ist, der außerhalb des gültigen Bereichs liegt.

Standardaktion: Die Task wird abnormal beendet.

51 NOPASSBKWR

Tritt auf, wenn kein Sparbuch vorhanden ist.

Standardaktion: Die Task wird abnormal beendet.

61 NOTALLOC

Tritt auf, falls die Anwendung nicht Eigner der im Befehl angegebenen Funktion ist.

Standardaktion: Die Task wird abnormal beendet.

81 TERMERR

Tritt für einen terminalbezogenen Fehler auf, wie zum Beispiel einen Sitzungsfehler. Diese Bedingung gilt nur für Terminals, die mit z/OS Communications Server verbunden sind.

Diese Bedingung kann durch eine Anforderung CANCEL TASK durch ein Knotenfehlerprogramm des Benutzers verursacht werden, falls für die Task eine Terminalsteuerungsanforderung aussteht, wenn der Sitzungsfehler durch das Programm des Knotens für abnormale Bedingungen (DFHZNAC) abgewickelt wird.

Standardaktion: Die Task wird abnormal mit dem Abbruchcode ATNI beendet.

03 WRBRK

Tritt auf, falls der Befehl durch die Abruftaste beendet wird.

Standardaktion: Die Bedingung wird ignoriert.

SEND CONTROL

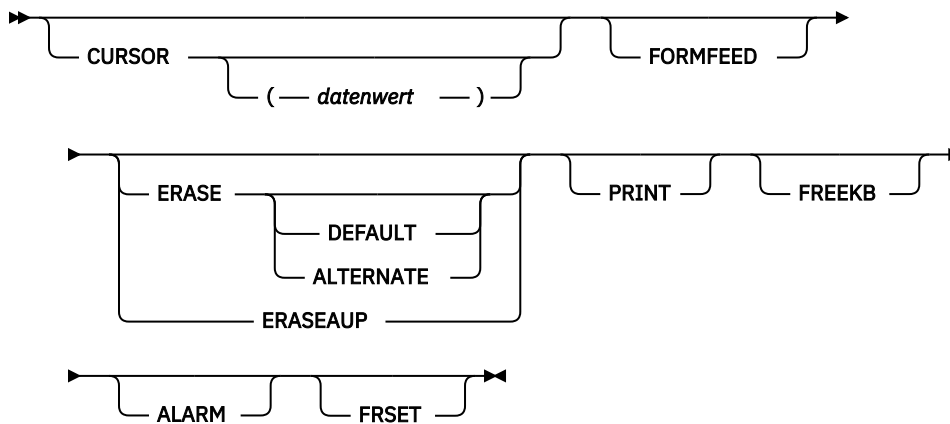
Sendet Einheitensteuerungen ohne Karten- oder Textdaten an ein Terminal. Die Schlüsselwörter werden unterteilt in diejenigen, die von minimaler, standardmäßiger und vollständiger BMS unterstützt werden.

Weitere Informationen zu BMS finden Sie unter [Basic Mapping Support](#).

SEND CONTROL

➡ SEND CONTROL ➡

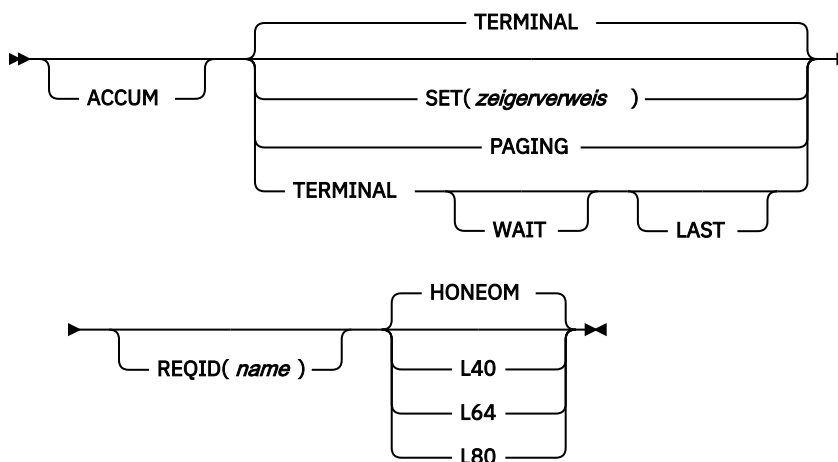
SEND CONTROL (minimale BMS)



SEND CONTROL (standardmäßige BMS)



SEND CONTROL (vollständige BMS)



Bedingungen: IGREQCD, IGREQID, INVLDC, INVPARTN, INVREQ, RETPAGE, TSIOERR, WRBRK

Beschreibung

Der Befehl SEND CONTROL sendet Einheitensteuerungen an ein Terminal.

Wenn Sie den Befehl SEND CONTROL mit einer der Optionen ALARM, FREEKB, FRSET, HONEOM, L40, L64, L80 oder PRINT verwenden, können Sie sich im Abschnitt zur [Option CTRL für DFHMDI](#) über die Optionspriorität informieren.

ACCUM

Gibt an, dass dieser Befehl zu einer Reihe von Befehlen gehört, die zur Erstellung einer logischen Nachricht verwendet werden. Die logische Nachricht wird durch einen Befehl SEND PAGE abgeschlossen oder durch einen Befehl PURGE MESSAGE gelöscht.

ACTPARTN(name)

Gibt den 1 bis 2 Zeichen langen Namen der zu aktivierenden Partition an. Durch das Aktivieren einer Partition wird der Cursor in die angegebene Partition verschoben und die Tastatur für die angegebene Partition entsperrt.

Falls das Zielterminal Partitionen nicht unterstützt oder keine Anwendungspartitionsgruppe festgelegt ist, wird diese Option ignoriert.

ALARM

Gibt an, dass die 3270-Funktion für akustische Signale aktiviert werden soll. Bei logischen Einheiten, die Funktionsverwaltungsheader unterstützen (mit Ausnahme von interaktiven logischen Einheiten und logischen Einheiten mit Stapel) wird BMS durch die Option ALARM angewiesen, die Markierung für Alarmsignale im Funktionsverwaltungsheader festzulegen.

ALTERNATE

Legt die Verwendung der alternativen Anzeigegröße durch das Terminal fest.

CURSOR(datenwert)

Gibt die Position an, an die der 3270- oder 3604-Cursor nach Abschluss eines Befehls SEND CONTROL zurückgesetzt werden soll.

Der Datenwert muss ein Halbwort-Binärwert sein, der die Cursorposition bezogen auf Null angibt; welcher Wertebereich angegeben werden kann, ist von der Größe der verwendeten Anzeige abhängig.

Bei Verwendung der Option ACCUM wird der Cursor anhand des zuletzt angegebenen Wertes von CURSOR positioniert.

In der Option CURSOR muss ein positiver Wert angegeben werden. Ein negativer Wert führt zu unvorhersehbaren Ergebnissen.

Falls diese Option weggelassen wird, wird der Cursor an Position Null der Anzeige platziert.

DEFAULT

Legt die Verwendung der Standardanzeigegröße durch das Terminal fest.

ERASE

Gibt an, dass der Anzeigendruckerpuffer oder die Partition gelöscht und der Cursor in die linke obere Ecke der Anzeige zurückgesetzt werden soll. (Diese Option gilt nur für die 3270- bzw. 8775-Anzeige und die 3604-Tastaturanzeige.)

Die erste Ausgabeoperation in einer Transaktion oder in einer Serie von Pseudodialogtransaktionen sollte immer die Option ERASE angeben. Für Transaktionen, die an 3270-Anzeigen oder -Drucker angehängt sind, stellt diese Option (sofern sie nicht explizit durch die Option DEFAULT oder ALTERNATE überschrieben wird) außerdem sicher, dass die richtige Anzeigegröße ausgewählt wird, die für die Transaktion durch die Option SCRNSIZE in der RDO-Definition PROFILE definiert ist.

ERASEAUP

Gibt an, dass alle ungeschützten Zeichenpositionen in der Partition oder der gesamten Anzeige gelöscht werden sollen. (Diese Option gilt nur für die 3270- bzw. 8775-Anzeige.)

FORMFEED

Gibt an, dass eine neue Seite erforderlich ist. Bei 3270-Druckern und -Anzeigen muss das FORMFEED-Zeichen am Anfang des Puffers positioniert sein. Das Anwendungsprogramm muss daher sicherstellen, dass diese Pufferposition nicht durch Masken- oder Textdaten überschrieben wird. Falls das Zielterminal die Option FORMFEED nicht unterstützt (also die RDO-Option FORMFEED für TYPE-TERM nicht verwendet wurde), wird diese Option ignoriert.

FREEKB

Gibt an, dass die 3270-Tastatur entsperrt werden soll. Wenn die Option FREEKB nicht angegeben wird, bleibt die Tastatur gesperrt.

Es ist zu beachten, dass der Status für die Tastatursperre auf einem Terminal, das Partitionen unterstützt, für jede Partition separat verwaltet wird.

FRSET

Gibt an, dass die Tags für modifizierte Daten (Modified Data Tags, MDTs) von allen Feldern, die sich gegenwärtig im Puffer für die 3270-Einheit (oder im Puffer für die Partition) befinden, auf die Bedingung für nicht modifizierte Daten zurückgesetzt werden sollen (also eine Feldzurücksetzung stattfinden soll).

Auf diese Weise kann der Operand ATTRB von DFHMDF für die nächste angeforderte Zuordnung den Endstatus der Felder steuern, die infolge eines BMS-Befehls geschrieben bzw. neu geschrieben werden, falls keine anderen Attributinformationen in die symbolische Zuordnung geschrieben wurden.

HONEOM

Gibt an, dass die Standardzeilenlänge des Druckers verwendet werden soll. Diese Länge sollte mit der Länge identisch sein, die durch die RDO-Optionen PAGESIZE oder ALTPAGE für TYPETERM angegeben ist.

LAST

Gibt an, dass es sich bei dieser Operation um die letzte Ausgabeoperation für eine Transaktion und daher um das Ende einer Klammer handelt. Diese Option gilt nur für logische Einheiten.

LDC(name)

Gibt ein zweistelliges mnemonisches Zeichen an, mit dessen Hilfe der logische Einheitencode (Logical Device Code, LDC) ermittelt wird, der im Funktionsverwaltungsheder an die logische Einheit übertragen werden soll. Das mnemonische Zeichen gibt einen LDC-Eintrag an, der durch ein Makro DFHTCT TYPE=LDC definiert wird. Bei Angabe eines logischen Einheitencodes verwendet BMS den Einheitentyp, die Seitengröße und den Seitenstatus, die dem logischen Einheitencode zugeordnet sind, um die Nachricht zu formatieren. Diese Werte werden aus der erweiterten lokalen LDC-Tabelle für die logische Einheit entnommen, falls sie eine solche Tabelle besitzt. Für den Fall, dass die logische Einheit nur eine lokale (nicht erweiterte) LDC-Tabelle besitzt, werden die Werte aus der LDC-Tabelle des Systems entnommen. Der numerische Wert des logischen Einheitencodes wird aus der lokalen LDC-Tabelle abgerufen, es sei denn, dass dies eine nicht erweiterte Tabelle ist und der Wert nicht angegeben ist. In diesem Fall wird er aus der Systemtabelle abgerufen.

L40, L64 oder L80

Gibt die Zeilenlänge für einen 3270-Drucker an; hierdurch wird eine Zeilenschaltung und ein Zeilenvorschub erzwungen, nachdem 40, 64 oder 80 Zeichen in einer Zeile gedruckt wurden. Wenn diese Einstellung von der Seitenbreite abweicht, die durch die RDO-Optionen PAGESIZE oder ALTPAGE für TYPETERM angegeben sind, sind mangelhafte Ergebnisse wahrscheinlich.

Bei Verwendung der Optionen können Sie sich im Abschnitt zur Option CRTL von DFHMDI über die Optionspriorität informieren.

MSR(datenwert)

Gibt den 4 Byte umfassenden Datenwert an, mit dem der 10/63-Magnetstreifenleser gesteuert wird, der an ein 8775- oder 3643-Terminal angeschlossen ist. In DFHMSRCA wird eine Reihe von Konstanten bereitgestellt, die das Festlegen dieses 4 Byte umfassenden Bereichs unterstützen. Eine vollständige Liste finden Sie unter „Steuerwertkonstanten für Magnetcodeleser (DFHMSRCA)“ auf Seite 845. Falls die RDO-Option MSRCONTROL für TYPETERM nicht verwendet wurde, wird diese Option ignoriert.

OUTPARTN(name)

Gibt den 1 bis 2 Zeichen langen Namen der Partition an, an die Daten gesendet werden sollen. Falls das Terminal Partitionen nicht unterstützt oder ihm keine Anwendungspartitionsgruppe zugeordnet ist, wird diese Option ignoriert. Wenn es eine Anwendungspartitionsgruppe gibt und die Option OUTPARTN nicht angegeben wird, werden die Daten an die Partition gesendet, die im Operanden PARTN der Maskendefinitionsmakros DFHMDS (siehe „DFHMDS“ auf Seite 871) oder DFHMDI (siehe „DFHMDI“ auf Seite 862) benannt ist. Falls keine Masken verwendet werden oder es keinen Operanden PARTN gibt, wird die Ausgabe an die erste Partition in der Partitionsgruppe gesendet.

PAGING

Gibt an, dass die Ausgabedaten nicht sofort an das Terminal gesendet, sondern im temporären Speicher abgelegt und als Antwort auf Blätterbefehle angezeigt werden sollen, die durch den Terminalbediener eingegeben werden.

Falls die Option PAGING mit einem Präfix REQID angegeben wird, das für Warteschlange für temporären Speicher definiert ist, die als wiederherstellbar definiert sind, stellt CICS die Nachrichtenwiederherstellung für logische Nachrichten bereit, falls die Task einen Synchronisationspunkt erreicht hat.

PRINT

Gibt an, dass auf einem 3270-Drucker oder einer 3275-Einheit mit Druckeradapterfunktion eine Druckoperation gestartet werden soll bzw. dass Daten auf einer Einheit des LU-Typs 2 (3274/76 oder 3790) auf einem durch den Controller zugeordneten Drucker gedruckt werden sollen. Falls diese Option nicht angegeben ist, werden die Daten an den Druckerpuffer gesendet, jedoch nicht gedruckt.

REQID(name)

Gibt ein zweistelliges Präfix an, das als Teil einer Kennung des temporären Speichers für die CICS-Nachrichtenwiederherstellung verwendet werden soll. Für jede logische Nachricht kann nur ein einziges Präfix angegeben werden. Das Standardpräfix ist **.

Die BMS-Nachrichtenwiederherstellung steht für eine logische Nachricht nur dann zur Verfügung, wenn die Option PAGING in den BMS-Befehlen SEND angegeben ist und wenn der Synchronisationspunkt erreicht wurde.

SET(zeigerverweis)

Gibt den Zeiger an, der auf die Adresse der Ausgabedaten gesetzt werden soll.

Die Option SET gibt an, dass fertiggestellte Seiten an das Anwendungsprogramm zurückgegeben werden sollen. Der Zeiger wird auf die Adresse einer Liste von fertiggestellten Seiten gesetzt.

Bei Angabe von TIOAPFX=YES in der Zuordnungsdefinition enthält der zurückgegebene Zeiger die Adresse des Präfix für den Ein-/Ausgabebereich des Terminals (Terminal Input Output Area, TIOA). Die Benutzerdaten beginnen an der relativen Position X'0C' ausgehend vom Anfang des TIOA-Präfix.

TERMINAL

Gibt an, dass die Ausgabedaten an das Terminal gesendet werden sollen, von dem die Transaktion stammt.

WAIT

Gibt an, dass die Steuerung erst nach Abschluss der Ausgabeoperation an das Anwendungsprogramm zurückgegeben werden soll.

Falls die Option WAIT nicht angegeben ist, wird die Steuerung an das Anwendungsprogramm zurückgegeben, sobald die Ausgabeoperation gestartet worden ist. Ein nachfolgender Eingabe- oder Ausgabebefehl (Terminalsteuerung, BMS oder Stapeldatenaustausch) führt dazu, dass sich das Anwendungsprogramm im Wartestatus befindet, bis der vorherige Befehl abgeschlossen wurde.

Bedingungen

57 IGREQCD

Tritt auf, wenn versucht wird, einen Befehl SEND CONTROL auszuführen, nachdem ein Datenflusssteuerungsbefehl SIGNAL mit einem Code für die Anforderung der Richtungsänderung (Request change direction, RCD) von einer logischen Einheit des LU-Typs 4 empfangen wurde.

Standardaktion: Die Task wird abnormal beendet.

39 IGREQID

Tritt auf, falls das in der Option REQID angegebene Präfix nicht mit dem von einer vorherigen Option REQID oder mit dem standardmäßig für diese logische Nachricht erstellten Präfix, also REQID (**), identisch ist.

Standardaktion: Die Task wird abnormal beendet.

41 INVLDLDC

Tritt auf, falls das angegebene mnemonische Zeichen für den logischen Einheitencode (LDC) nicht in der LDC-Liste für die logische Einheit enthalten ist.

Standardaktion: Die Task wird abnormal beendet.

65 INVPARTN

Tritt auf, falls die angegebene Partition nicht in der Partitionsgruppe definiert ist, die dem Anwendungsprogramm zugeordnet ist.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

RESP2-Werte:

200

Eine Serveranwendung für Distributed Program Link hat versucht, Daten in ihrer Sitzung für die Funktionsverlagerung (ihrer Hauptfunktion) zu senden.

Diese Bedingung (RESP2 nicht festgelegt) tritt auch in der folgenden Situation auf:

- Steuerinformationen werden in derselben Partition oder in demselben logischen Einheitencode wie zugeordnete Daten ausgegeben, während eine logische Nachricht von BMS aktiv ist. Falls weder Partitionen noch logische Einheitencodes verwendet werden, werden die Steuerinformationen in derselben Einheit als zugeordnete Daten ausgegeben.

Standardaktion: Die Task wird abnormal beendet.

32 RETPAGE

Tritt auf, falls die Option SET angegeben ist und eine fertiggestellte Seite für die Rückgabe an das Anwendungsprogramm bereit ist.

Standardaktion: Die Steuerung wird unmittelbar nach dem BMS-Befehl SEND an das Anwendungsprogramm zurückgegeben.

35 TSIOERR

Tritt auf, falls ein nicht behebbarer Ein-/Ausgabefehler für den temporären Speicher aufgetreten ist.

Standardaktion: Die Task wird abnormal beendet.

03 WRBRK

Tritt auf, falls der Befehl unterbrochen wurde, weil der Terminalbediener die Abruftaste gedrückt hat. Diese Bedingung gilt nur für das Kommunikationsterminal IBM 2741 und nur dann, wenn die Unterbrechung des Schreibvorgangs für CICS unterstützt wird.

Standardaktion: Die Bedingung wird ignoriert.

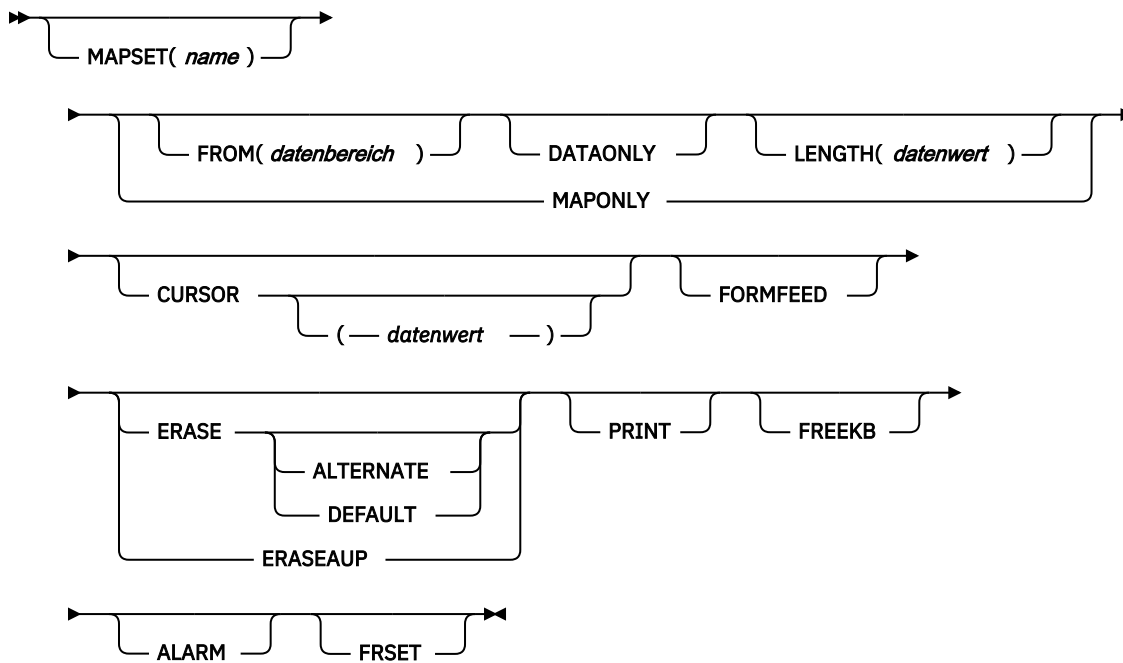
SEND MAP

Sendet zugeordnete Ausgabedaten an ein Terminal. Die Schlüsselwörter werden unterteilt in diejenigen, die von minimaler, standardmäßiger und vollständiger BMS unterstützt werden. Weitere Informationen zu BMS finden Sie unter [Basic Mapping Support](#).

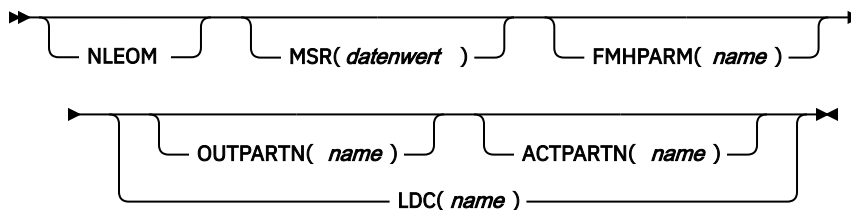
SEND MAP

➤ SEND MAP(*name*) ➤

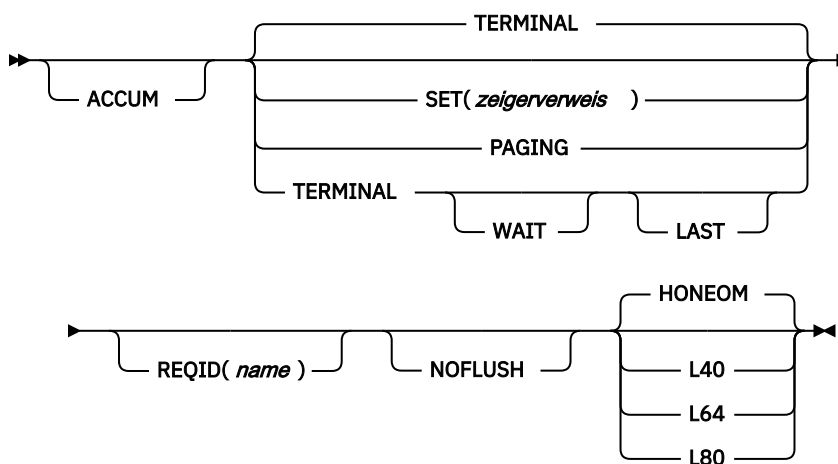
SEND MAP (minimale BMS)



SEND MAP (standardmäßige BMS)



SEND MAP (vollständige BMS)



Bedingungen: IGREQCD, IGREQID, INVLDC, INVMPsz, INVPARTN, INVREQ, OVERFLOW, RETPAGE, TSIOERR, WRBRK

Beschreibung

Der Befehl SEND MAP sendet Ausgabedaten an ein Terminal.

Wenn Sie den Befehl SEND MAP mit einer der Optionen ALARM, FREEKB, FRSET, HONEOM, L40, L64, L80 oder PRINT verwenden, können Sie sich im Abschnitt „DFHMDI“ auf Seite 862 über die Optionspriorität informieren.

Die Maskendefinition finden Sie im Abschnitt BMS-Makros.

Optionen

ACCUM

Gibt an, dass dieser Befehl zu einer Reihe von Befehlen gehört, die zur Erstellung einer logischen Nachricht verwendet werden. Die logische Nachricht wird durch einen Befehl SEND PAGE abgeschlossen oder durch einen Befehl PURGE MESSAGE gelöscht.

ACTPARTN(name)

Gibt den 1 bis 2 Zeichen langen Namen der zu aktivierenden Partition an. Durch das Aktivieren einer Partition wird der Cursor in die angegebene Partition verschoben und die Tastatur für die angegebene Partition entsperrt.

Falls das Zielterminal Partitionen nicht unterstützt oder keine Anwendungspartitionsgruppe festgelegt ist, wird diese Option ignoriert.

ALARM

Gibt an, dass die 3270-Funktion für akustische Signale aktiviert werden soll. Bei logischen Einheiten, die Funktionsverwaltungsheader unterstützen (mit Ausnahme von interaktiven logischen Einheiten und logischen Einheiten mit Stapel) wird BMS durch die Option ALARM angewiesen, die Markierung für Alarmsignale im Funktionsverwaltungsheader festzulegen.

Bei Verwendung der Option ALARM können Sie sich im Abschnitt „DFHMDI“ auf Seite 862 über die Optionspriorität informieren.

ALTERNATE

Legt die Verwendung der alternativen Anzeigegröße durch das Terminal fest.

CURSOR(datenwert)

Gibt die Position an, an die der 3270- oder 3604-Cursor nach Abschluss eines Befehls SEND MAP zurückgesetzt werden soll.

Der Datenwert muss ein Halbwort-Binärwert sein, der die Cursorposition bezogen auf Null angibt; welcher Wertebereich angegeben werden kann, ist von der Größe der verwendeten Anzeige abhängig. Falls kein Datenwert angegeben wird, wird die symbolische Cursorpositionierung angenommen.

Diese Option setzt jede Option IC des Operanden ATTRB von DFHMDF außer Kraft. Bei Verwendung der Option ACCUM wird der Cursor anhand des zuletzt angegebenen Wertes von CURSOR positioniert.

In der Option CURSOR muss ein positiver Wert angegeben werden. Ein negativer Wert führt zu unvorhersehbaren Ergebnissen.

DATAONLY

Gibt an, dass ausschließlich Anwendungsprogrammdateien geschrieben werden sollen. Für jedes Feld in den bereitgestellten Daten müssen die Attributzeichen (nur 3270) angegeben sein. Falls das Attributbyte in den vom Benutzer bereitgestellten Daten auf X'00' gesetzt ist, wird das Attributbyte in der Anzeige nicht geändert. Standarddaten oder Attribute aus der Zuordnung werden ignoriert.

DEFAULT

Legt die Verwendung der Standardanzeigegröße durch das Terminal fest.

ERASE

Gibt an, dass der Anzeigendruckerpuffer oder die Partition gelöscht und der Cursor in die linke obere Ecke der Anzeige zurückgesetzt werden soll. (Diese Option gilt nur für die 3270- bzw. 8775-Anzeige und die 3604-Tastaturanzeige.)

Die erste Ausgabeoperation in einer Transaktion oder in einer Serie von Pseudodialogtransaktionen sollte immer die Option ERASE angeben. Für Transaktionen, die an 3270-Anzeigen oder -Drucker angehängt sind, stellt diese Option (sofern sie nicht explizit durch die Option DEFAULT oder ALTERNATE

überschrieben wird) außerdem sicher, dass die richtige Anzeigegröße ausgewählt wird, die für die Transaktion durch die Option SCRNSIZE in der RDO-Definition PROFILE definiert ist.

ERASEUP

Gibt an, dass alle ungeschützten Zeichenpositionen in der Partition oder der gesamten Anzeige gelöscht werden sollen, bevor diese Seite der Ausgabe angezeigt wird. (Diese Option gilt nur für die 3270- bzw. 8775-Anzeige.)

FMHPARM(name)

Gibt den 1 bis 8 Zeichen langen Namen der zu verwendenden externen Zuordnung an. (Diese Option gilt nur für logische 3650-Einheiten mit externer Formatierung.)

FORMFEED

Gibt an, dass eine neue Seite erforderlich ist. Bei 3270-Druckern und -Anzeigen muss das FORMFEED-Zeichen am Anfang des Puffers positioniert sein. Das Anwendungsprogramm muss daher sicherstellen, dass diese Pufferposition nicht durch Masken- oder Textdaten überschrieben wird. Falls das Zielterminal die Option FORMFEED nicht unterstützt (also die RDO-Option FORMFEED für TYPE-TERM nicht verwendet wurde), wird diese Option ignoriert.

FREEKB

Gibt an, dass die 3270-Tastatur entsperrt werden soll, nachdem die Daten geschrieben wurden. Wenn die Option FREEKB nicht angegeben wird, bleibt die Tastatur gesperrt.

Es ist zu beachten, dass der Status für die Tastatursperre auf einem Terminal, das Partitionen unterstützt, für jede Partition separat verwaltet wird.

Bei Verwendung der Option FREEKB können Sie sich im Abschnitt zur Option CRTL von DFHMDI über die Optionspriorität informieren.

FROM(datenbereich)

Gibt den Datenbereich an, in dem die Daten enthalten sind, die verarbeitet werden sollen. Falls dieses Feld nicht angegeben ist, wird als Name standardmäßig der Name der Zuordnung mit dem Suffix 0 verwendet. Diese Angabe schließt das 12 Byte umfassende Präfix ein, das durch die Option TIO-APFX=YES für die BMS-Zuordnungsdefinitionen DFHMDI und DFHMSD generiert wird (siehe hierzu den Abschnitt „DFHMDI“ auf Seite 862 und den Abschnitt über die Angabe von NODDS im BMS-Operanden).

FRSET

Gibt an, dass die Tags für modifizierte Daten (Modified Data Tags, MDTs) von allen Feldern, die sich gegenwärtig im Puffer für die 3270-Einheit (oder im Puffer für die Partition) befinden, auf die Bedingung für nicht modifizierte Daten zurückgesetzt werden sollen (also eine Feldzurücksetzung stattfinden soll), bevor Zuordnungsdaten in den Puffer geschrieben werden.

Auf diese Weise kann der Operand ATTRB von DFHMDI für die angeforderte Zuordnung den Endstatus der Felder steuern, die infolge eines BMS-Befehls geschrieben bzw. neu geschrieben werden, falls keine anderen Attributinformationen in die symbolische Zuordnung geschrieben wurden.

Bei Verwendung der Option FRSET können Sie sich im Abschnitt „DFHMDI“ auf Seite 862 über die Optionspriorität informieren.

HONEOM

Gibt an, dass die Standardzeilenlänge des Druckers verwendet werden soll. Diese Länge sollte mit der Länge, die durch die RDO-Optionen PAGESIZE oder ALTPAGE für TYPETERM angegeben ist, sowie mit der Breite der Druckerwalze identisch sein; andernfalls werden die Daten möglicherweise nicht korrekt formatiert.

Bei Verwendung der Option HONEOM können Sie sich im Abschnitt „DFHMDI“ auf Seite 862 über die Optionspriorität informieren.

LAST

Gibt an, dass es sich bei dieser Operation um die letzte Ausgabeoperation für eine Transaktion und daher um das Ende einer Klammer handelt. Diese Option gilt nur für logische Einheiten.

LDC(name)

Gibt ein zweistelliges mnemonisches Zeichen an, mit dessen Hilfe der logische Einheitencode (Logical Device Code, LDC) ermittelt wird, der im Funktionsverwaltungshdr an die logische Einheit übertragen werden soll. Das mnemonische Zeichen gibt einen LDC-Eintrag an, der durch ein Makro DFHTCT TYPE=LDC definiert wird.

Bei Angabe eines logischen Einheitencodes verwendet BMS den Einheitentyp, die Seitengröße und den Seitenstatus, die dem logischen Einheitencode zugeordnet sind, um die Nachricht zu formatieren. Diese Werte werden aus der erweiterten lokalen LDC-Tabelle für die logische Einheit entnommen, falls sie eine solche Tabelle besitzt. Für den Fall, dass die logische Einheit nur eine lokale (nicht erweiterte) LDC-Tabelle besitzt, werden die Werte aus der LDC-Tabelle des Systems entnommen. Der numerische Wert des logischen Einheitencodes wird aus der lokalen LDC-Tabelle abgerufen, es sei denn, dass dies eine nicht erweiterte Tabelle ist und der Wert nicht angegeben ist. In diesem Fall wird er aus der Systemtabelle abgerufen.

Falls die Option LDC nicht angegeben ist, wird das im Makro DFHMSD angegebene mnemonische Zeichen für den logischen Einheitencode verwendet; weitere Details enthält der Abschnitt „DFHMSD“ auf Seite 871. Wenn die Option LDC auch im Makro DFHMSD nicht angegeben ist, richtet sich die Aktion wie folgt nach dem Typ der logischen Einheit:

3601, logische Einheit

Es wird der erste Eintrag in einer lokalen oder einer erweiterten lokalen LDC-Tabelle verwendet, wenn es eine solche Tabelle gibt. Falls auf diese Weise kein Standardwert abgerufen werden kann, wird der numerische Wert für einen leeren LDC (X'00') verwendet. Als Seitengröße wird der Wert verwendet, der in den RDO TYPETERM-Optionen PAGESIZE oder ALTPAGE angegeben ist, bzw. der Wert (1,40), falls diese Optionen keinen Wert angeben.

Logische Einheit mit LU-Typ 4, logische Einheit mit Stapel oder logische Einheit mit Stapeldatenaustausch

Es wird nicht die lokale LDC-Tabelle für die Bereitstellung eines Standard-LDC verwendet. Stattdessen wird die Nachricht an die Konsole der logischen Einheit übertragen (also an jedes beliebige Medium, das die logische Einheit für den Empfang solcher Nachrichten wählt). Bei einer logischen Einheit mit Stapeldatenaustausch impliziert dies nicht das Senden eines logischen Einheitencodes im Funktionsverwaltungshdr. Die Seitengröße wird auf dieselbe Weise wie bei einer logischen 3601-Einheit abgerufen.

LENGTH(datenwert)

Gibt die Länge der zu formatierenden Daten in Form eines Halbwort-Binärwertes an.

Falls der Datenbereich für das Senden der Zuordnung größer als die zuzuordnenden Daten ist, sollte die Option LENGTH angegeben werden. Diese Angabe sollte das 12 Byte umfassende Präfix einschließen, das durch die Option TIOAPFX=YES für die BMS-Zuordnungsdefinitionen DFHMDI und DFHMSD generiert wird (siehe hierzu den Abschnitt „DFHMDI“ auf Seite 862 und den Abschnitt über die Angabe von NODDS im BMS-Operanden). Eine Beschreibung einer sicheren Obergrenze finden Sie unter „LENGTH-Optionen in CICS-Befehlen“ auf Seite 10.

L40, L64 oder L80

Gibt die Zeilenlänge für einen 3270-Drucker an; hierdurch wird eine Zeilenschaltung und ein Zeilenvorschub erzwungen, nachdem 40, 64 oder 80 Zeichen in einer Zeile gedruckt wurden. Wenn diese Einstellung von der Seitenbreite abweicht, die durch die RDO-Optionen PAGESIZE oder ALTPAGE für TYPETERM angegeben sind, sind mangelhafte Ergebnisse wahrscheinlich.

Bei Verwendung der Optionen können Sie sich im Abschnitt „DFHMDI“ auf Seite 862 über die Optionspriorität informieren.

MAP(name)

Gibt den 1 bis 7 Zeichen langen Namen der zu verwendenden Zuordnung an.

MAPONLY

Gibt an, dass nur Standarddaten aus der Zuordnung geschrieben werden sollen.

MAPSET(name)

Gibt den 1 bis 7 Zeichen langen Namen ohne Suffix der zu verwendenden Maskengruppe an. Die Maskengruppe muss sich in der CICS-Programmbibliothek befinden. Sie kann entweder mithilfe der On-

lineresourcendefinition (RDO) oder durch die automatische Programminstallation bei ihrer erstmaligen Verwendung definiert werden. Falls diese Option nicht angegeben wird, wird der in der Option MAP angegebene Name als Name der Maskengruppe angenommen.

Die Anzahl der Zuordnungen pro Maskengruppe ist auf maximal 9998 begrenzt.

MSR(datenwert)

Gibt den 4 Byte umfassenden Datenwert an, mit dem der 10/63-Magnetstreifenleser gesteuert wird, der an ein 8775- oder 3643-Terminal angeschlossen ist. In DFHMSRCA wird eine Reihe von Konstanten bereitgestellt, die das Festlegen dieses 4 Byte umfassenden Bereichs unterstützen. Eine vollständige Liste finden Sie unter „Steuerwertkonstanten für Magnetcodeleser (DFHMSRCA)“ auf Seite 845. Falls die RDO-Option MSRCONTROL für TYPETERM nicht verwendet wurde, wird diese Option ignoriert.

NLEOM

Gibt an, dass Daten für einen 3270-Drucker oder eine 3275-Anzeige mit Druckeradapterfunktion mit Leerzeichen und Zeilenvorschubzeichen erstellt werden sollen und dass am Ende der Daten ein Zeichen für das Nachrichtenende eingefügt werden soll. Beim Drucken der Daten bewirkt jedes Zeilenvorschubzeichen, dass die Druckausgabe in der nächsten Zeile fortgesetzt wird, und das Zeichen für das Nachrichtenende beendet die Druckausgabe.

Diese Option muss im ersten Befehl SEND MAP angegeben sein, der zum Erstellen einer logischen Nachricht verwendet wird. Die Option wird ignoriert, falls die Einheit, die die Nachricht (direkt oder weitergeleitet) empfängt, keine der oben genannten Einheiten ist.

Falls diese Option verwendet wird, ist weder eine Pufferaktualisierung noch eine Attributänderung von zuvor in den Puffer geschriebenen Feldern zulässig. CICS bezieht bei jedem Schreibvorgang für das Terminal die Option ERASE ein.

Das Zeilenvorschubzeichen belegt eine Pufferposition. Eine Reihe von Pufferpositionen, deren Anzahl dem Wert der RDO-Optionen PAGESIZE oder ALTPAGE von TYPETERM für dieses Terminal entspricht, sind für Daten nicht verfügbar. Dies kann dazu führen, dass Daten im Puffer umlaufen; in einem solchen Fall muss der Wert für PAGESIZE oder ALTPAGE herabgesetzt werden.

Die Option NLEOM setzt die Option ALARM außer Kraft, falls sie angegeben ist.

NOFLUSH

Gibt an, dass CICS Seiten nach dem Abschluss nicht löscht, sondern die Steuerung an das Programm zurückgibt (Option OVERFLOW ist in EIBRESP festgelegt).

OUTPARTN(name)

Gibt den 1 bis 2 Zeichen langen Namen der Partition an, an die Daten gesendet werden sollen. Falls das Terminal Partitionen nicht unterstützt oder ihm keine Anwendungspartitionsgruppe zugeordnet ist, wird diese Option ignoriert. Wenn es eine Anwendungspartitionsgruppe gibt und die Option OUTPARTN nicht angegeben wird, werden die Daten an die Partition gesendet, die im Operanden PARTN der Zuordnungsdefinitionen DFHMSD oder DFHMDI benannt ist. Falls keine Zuordnungen verwendet werden oder es keinen Operanden PARTN gibt, wird die Ausgabe an die erste Partition in der Partitionsgruppe gesendet.

PAGING

Gibt an, dass die Ausgabedaten nicht sofort an das Terminal gesendet, sondern im temporären Speicher abgelegt und als Antwort auf Blätterbefehle angezeigt werden sollen, die durch den Terminalbediener eingegeben werden.

Falls die Option PAGING mit einem Präfix REQID angegeben wird, das für Warteschlange für temporären Speicher definiert ist, die als wiederherstellbar definiert sind, stellt CICS die Nachrichtenwiederherstellung für logische Nachrichten bereit, falls die Task einen Synchronisationspunkt erreicht hat.

PRINT

Gibt an, dass auf einem 3270-Drucker oder einer 3275-Einheit mit Druckeradapterfunktion eine Druckoperation gestartet werden soll bzw. dass Daten auf einer Einheit des LU-Typs 2 (3274/76 oder 3790) auf einem durch den Controller zugeordneten Drucker gedruckt werden sollen. Falls diese Option nicht angegeben ist, werden die Daten an den Druckerpuffer gesendet, jedoch nicht gedruckt.

Bei Verwendung der Option PRINT können Sie sich im Abschnitt „DFHMDI“ auf Seite 862 über die Optionspriorität informieren.

REQID(name)

Gibt ein zweistelliges Präfix an, das als Teil einer Kennung des temporären Speichers für die CICS-Nachrichtenwiederherstellung verwendet werden soll. Für jede logische Nachricht kann nur ein einziges Präfix angegeben werden. Das Standardpräfix ist **.

Die BMS-Nachrichtenwiederherstellung steht für eine logische Nachricht nur dann zur Verfügung, wenn die Option PAGING in den BMS-Befehlen SEND angegeben ist und wenn der Synchronisationspunkt erreicht wurde.

SET(zeigerverweis)

Gibt den Zeiger an, der auf die Adresse der Eingabe- oder Ausgabedaten gesetzt werden soll.

Die Option SET gibt an, dass fertiggestellte Seiten an das Anwendungsprogramm zurückgegeben werden sollen. Der Zeiger wird auf die Adresse einer Liste von fertiggestellten Seiten gesetzt.

Das Anwendungsprogramm erhält die Steuerung entweder sofort nach dem Befehl SEND MAP zurück (falls die aktuelle Seite noch nicht fertiggestellt ist) oder aber bei dem Kennsatz, der in einem Befehl HANDLE CONDITION RETPAGE angegeben ist (falls die Seite fertiggestellt wurde).

Bei Angabe von TIOAPFX=YES in der Zuordnungsdefinition enthält der zurückgegebene Zeiger die Adresse des Präfix für den Ein-/Ausgabebereich des Terminals (Terminal Input Output Area, TIOA). Die Benutzerdaten beginnen an der relativen Position X'0C' ausgehend vom Anfang des TIOA-Präfix.

TERMINAL

Gibt an, dass die Ausgabedaten an das Terminal gesendet werden sollen, von dem die Transaktion stammt.

WAIT

Gibt an, dass die Steuerung erst nach Abschluss der Ausgabeoperation an das Anwendungsprogramm zurückgegeben werden soll.

Falls die Option WAIT nicht angegeben ist, wird die Steuerung an das Anwendungsprogramm zurückgegeben, sobald die Ausgabeoperation gestartet worden ist. Ein nachfolgender Eingabe- oder Ausgabebefehl (Terminalsteuerung, BMS oder Stapeldatenaustausch) führt dazu, dass sich das Anwendungsprogramm im Wartestatus befindet, bis der vorherige Befehl abgeschlossen wurde.

Bedingungen

Einige der folgenden Bedingungen können kombiniert auftreten. Falls mehrere Bedingungen auftreten, wird nur die erste Bedingung an das Anwendungsprogramm übergeben.

57 IGREQCD

Tritt auf, wenn versucht wird, einen Befehl SEND MAP auszuführen, nachdem ein Datenflusssteuerungsbefehl SIGNAL mit einem Code für die Anforderung der Richtungsänderung (Request change direction, RCD) von einer logischen Einheit des LU-Typs 4 empfangen wurde.

Standardaktion: Die Task wird abnormal beendet.

39 IGREQID

Tritt auf, falls das in der Option REQID angegebene Präfix nicht mit dem von einer vorherigen Option REQID oder mit dem standardmäßig für diese logische Nachricht erstellten Präfix, also REQID (**), identisch ist.

Standardaktion: Die Task wird abnormal beendet.

41 INVLDC

Tritt auf, falls das angegebene mnemonische Zeichen für den logischen Einheitencode (LDC) nicht in der LDC-Liste für die logische Einheit enthalten ist.

Standardaktion: Die Task wird abnormal beendet.

38 INVMP SZ

Tritt auf, falls die angegebene Zuordnung zu breit für das Terminal ist oder falls ein Befehl HANDLE CONDITION OVERFLOW aktiv und die angegebene Zuordnung zu lang für das Terminal ist.

Standardaktion: Die Task wird abnormal beendet.

65 INVPARTN

Tritt auf, falls die angegebene Partition nicht in der Partitionsgruppe definiert ist, die dem Anwendungsprogramm zugeordnet ist.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

RESP2-Werte:

200

Der Befehl ist für ein Serverprogramm von Distributed Program Link nicht zulässig.

Diese Bedingung tritt außerdem in einer der folgenden Situationen auf (RESP2 nicht festgelegt):

- Textdaten werden in derselben Partition oder in demselben logischen Einheitencode wie zugeordnete Daten ausgegeben, während eine logische Nachricht von BMS aktiv ist. Falls weder Partitionen noch logische Einheitencodes verwendet werden, werden Textdaten in derselben logischen Nachricht wie zugeordnete Daten ausgegeben.
- Ein separater Befehl SEND MAP mit der Option ACCUM wurde für das Terminal ausgegeben, von dem die Transaktion stammte, während eine weitergeleitete logische Nachricht erstellt wird.
- Ein Befehl SEND MAP wurde für eine Zuordnung ohne Feldspezifikationen unter Angabe der Option FROM ohne die Option DATAONLY ausgegeben.
- Während der Überlaufverarbeitung werden Daten an einen anderen logischen Einheitencode als dem Code gesendet, der den Seitenüberlauf verursacht hat.
- Es werden Partitionen verwendet und die Option OUTPARTN wurde im Befehl SEND MAP angegeben, aber in der Maskengruppendefinition wurde der Operand PARTN codiert. Falls diese Bedingung auftritt, deutet sie darauf hin, dass verschiedene Versionen der Maskengruppe unterschiedliche Werte für PARTN aufweisen und dass das für die Partition abgeleitete Suffix nicht mit dem Suffix der geladenen Maskengruppe identisch ist.
- Ein Befehl SEND MAP mit der Option DATAONLY wurde mit einem Datenbereich ausgegeben, der vom Benutzer angegeben wurde und sich oberhalb der 16-MB-Grenze befindet. Die Länge dieses Datenbereichs ist jedoch nicht länger als das TIOA-Präfix.

Standardaktion: Die Task wird abnormal beendet.

40 OVERFLOW

Tritt auf, falls die zugeordneten Daten nicht auf die aktuelle Seite passen. Diese Bedingung tritt nur dann auf, wenn ein Befehl HANDLE CONDITION OVERFLOW aktiv ist.

Standardaktion: Die Bedingung wird ignoriert.

32 RETPAGE

Tritt auf, falls die Option SET angegeben ist und eine fertiggestellte Seite für die Rückgabe an das Anwendungsprogramm bereit ist.

Standardaktion: Die Steuerung wird unmittelbar nach dem BMS-Befehl SEND MAP an das Anwendungsprogramm zurückgegeben.

35 TSIOERR

Tritt auf, falls ein nicht behebbarer Ein-/Ausgabefehler für den temporären Speicher aufgetreten ist.

Standardaktion: Die Task wird abnormal beendet.

03 WRBRK

Tritt auf, falls ein Befehl SEND MAP unterbrochen wurde, weil der Terminalbediener die Abruftaste gedrückt hat. Diese Bedingung gilt nur für das Kommunikationsterminal IBM 2741 und nur dann, wenn die Unterbrechung des Schreibvorgangs für CICS unterstützt wird.

Standardaktion: Die Bedingung wird ignoriert.

SEND MAP MAPPINGDEV

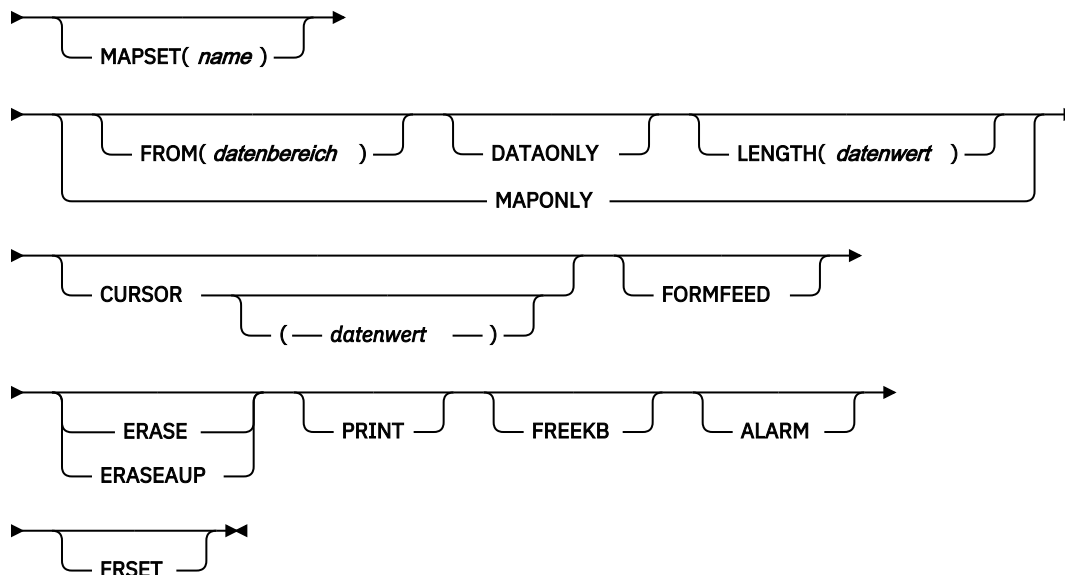
Erstellt zugeordnete Ausgabedaten, die später an ein mit MAPPINGDEV beschriebenes Terminal gesendet werden sollen.

Weitere Informationen zu BMS finden Sie unter [Basic Mapping Support](#).

Minimale BMS:

SEND MAP MAPPINGDEV

➔ SEND MAP(*name*) — MAPPINGDEV(*datenwert*) — SET(*zeigerverweis*) ➔



Bedingungen: INVMPsz, INVREQ

Beschreibung

Der Befehl SEND MAP MAPPINGDEV erstellt zugeordnete Ausgabedaten, die an ein Terminal gesendet werden sollen, das nicht die Hauptfunktion der Transaktion ist. Die zu verwendenden Terminalmerkmale werden durch MAPPINGDEV definiert.

Die zugeordneten Daten werden nicht übertragen, sondern in einem durch die Option SET definierten Puffer an die Anwendung zurückgegeben.

Optionen

ALARM

Gibt an, dass die 3270-Funktion für akustische Signale aktiviert werden soll. Bei logischen Einheiten, die Funktionsverwaltungsheader unterstützen (mit Ausnahme von interaktiven logischen Einheiten und logischen Einheiten mit Stapel) wird BMS durch die Option ALARM angewiesen, die Markierung für Alarmsignale im Funktionsverwaltungsheader festzulegen.

Bei Verwendung der Option ALARM können Sie sich im Abschnitt zur [Option CRTL von DFHMDI](#) über die Optionspriorität informieren.

CURSOR(*datenwert*)

Gibt die Position an, an die der 3270-Cursor nach Abschluss eines Befehls SEND MAP MAPPINGDEV zurückgesetzt werden soll.

Der Datenwert muss ein Halbwort-Binärwert sein, der die Cursorposition bezogen auf Null angibt; welcher Wertebereich angegeben werden kann, ist von der Größe der verwendeten Anzeige abhängig. Falls kein Datenwert angegeben wird, wird die symbolische Cursorpositionierung angenommen.

Diese Option setzt jede Option IC des Operanden ATTRB von DFHMDF außer Kraft.

In der Option CURSOR muss ein positiver Wert angegeben werden. Ein negativer Wert führt zu unvorhersehbaren Ergebnissen.

DATAONLY

Gibt an, dass ausschließlich Anwendungsprogrammdateien geschrieben werden sollen. Für jedes Feld in den bereitgestellten Daten müssen die Attributzeichen (nur 3270) angegeben sein. Falls das Attributbyte in den vom Benutzer bereitgestellten Daten auf X'00' gesetzt ist, wird das Attributbyte in der Anzeige nicht geändert. Standarddaten oder Attribute aus der Zuordnung werden ignoriert.

ERASE

Gibt an, dass der Anzeigendruckerpuffer gelöscht und der Cursor in die linke obere Ecke der Anzeige zurückgesetzt werden soll. (Diese Option gilt nur für die 3270- bzw. 8775-Anzeige.)

Die erste Ausgabeoperation in einer Transaktion oder in einer Serie von Pseudodialogtransaktionen sollte immer die Option ERASE angeben. Für Transaktionen, die an 3270-Anzeigen oder -Drucker angehängt sind, stellt diese Option außerdem sicher, dass die richtige Anzeigegröße ausgewählt wird, die für die Transaktion durch die Option SCRNSIZE in der RDO-Definition PROFILE definiert ist.

ERASEAUP

Gibt an, dass alle ungeschützten Zeichenpositionen in der gesamten Anzeige gelöscht werden sollen, bevor diese Seite der Ausgabe angezeigt wird. (Diese Option gilt nur für die 3270- bzw. 8775-Anzeige.)

FORMFEED

Gibt an, dass eine neue Seite erforderlich ist. Bei 3270-Druckern und -Anzeigen muss das FORMFEED-Zeichen am Anfang des Puffers positioniert sein. Das Anwendungsprogramm muss daher sicherstellen, dass diese Pufferposition nicht durch Masken- oder Textdaten überschrieben wird. Falls das Zielterminal die Option FORMFEED nicht unterstützt (also die RDO-Option FORMFEED für TYPE-TERM nicht verwendet wurde oder die Terminalsteuertabelle TYPE=TERMINAL nicht FF=YES angibt), wird diese Option ignoriert.

FREEKB

Gibt an, dass die 3270-Tastatur entsperrt werden soll, nachdem die Daten geschrieben wurden. Wenn die Option FREEKB nicht angegeben wird, bleibt die Tastatur gesperrt.

Bei Verwendung der Option FREEKB können Sie sich im Abschnitt zur [Option CRTL von DFHMDF](#) über die Optionspriorität informieren.

FROM(datenbereich)

Gibt den Datenbereich an, in dem die Daten enthalten sind, die verarbeitet werden sollen. Falls dieses Feld nicht angegeben ist, wird als Name standardmäßig der Name der Zuordnung mit dem Suffix 0 verwendet. Diese Angabe schließt das 12 Byte umfassende Präfix ein, das durch die Option TIO-APFX=YES für die BMS-Zuordnungsdefinitionen DFHMDF und DFHMDF generiert wird (siehe hierzu die Abschnitte über den [DFHMDF-Operanden TERM](#) und die [Angabe von NODDS im BMS-Operanden](#)).

FRSET

Gibt an, dass die Tags für modifizierte Daten (Modified Data Tags, MDTs) von allen Feldern, die sich gegenwärtig im Puffer für die 3270-Einheit befinden, auf die Bedingung für nicht modifizierte Daten zurückgesetzt werden sollen (also eine Feldzurücksetzung stattfinden soll), bevor Zuordnungsdaten in den Puffer geschrieben werden.

Auf diese Weise kann der Operand ATTRB von DFHMDF für die angeforderte Zuordnung den Endstatus der Felder steuern, die infolge eines BMS-Befehls geschrieben bzw. neu geschrieben werden, falls keine anderen Attributinformationen in die symbolische Zuordnung geschrieben wurden.

Bei Verwendung der Option FRSET können Sie sich im Abschnitt zur [Option CRTL von DFHMDF](#) über die Optionspriorität informieren.

LENGTH(datenwert)

Gibt die Länge der zu formatierenden Daten in Form eines Halbwort-Binärwertes an.

Falls der Datenbereich für das Senden der Zuordnung größer als die zuzuordnenden Daten ist, sollte die Option LENGTH angegeben werden. Diese Angabe sollte das 12 Byte umfassende Präfix einschließen, das durch die Option TIOAPFX=YES für die BMS-Zuordnungsdefinitionen DFHMDI und DFHMSD generiert wird (siehe hierzu die Abschnitte über den DFHMDI-Operanden TERM und die Angabe von NODDS im BMS-Operanden). Eine Beschreibung einer sicheren Obergrenze finden Sie unter „LENGTH-Optionen in CICS-Befehlen“ auf Seite 10.

MAP(name)

Gibt den 1 bis 7 Zeichen langen Namen der zu verwendenden Zuordnung an.

MAPPINGDEV(datenwert)

Gibt den Namen eines 3270-Terminals an, dessen BMS-Merkmale mit denen des Terminals übereinstimmen, an das die Daten letztlich mit einem Befehl SEND TEXT MAPPED bzw. einer Terminalsteuerung SEND oder CONVERSE gesendet werden.

MAPONLY

Gibt an, dass nur Standarddaten aus der Zuordnung geschrieben werden sollen.

MAPSET(name)

Gibt den 1 bis 7 Zeichen langen Namen ohne Suffix der zu verwendenden Maskengruppe an. Die Maskengruppe muss sich in der CICS-Programmbibliothek befinden. Sie kann entweder mithilfe der Onlineresourcendefinition (RDO) oder durch die automatische Programminstallation bei ihrer erstmaligen Verwendung definiert werden. Falls diese Option nicht angegeben wird, wird der in der Option MAP angegebene Name als Name der Maskengruppe angenommen.

Die Anzahl der Zuordnungen pro Maskengruppe ist auf maximal 9998 begrenzt.

PRINT

Gibt an, dass auf einem 3270-Drucker oder einer 3275-Einheit mit Druckeradapterfunktion eine Druckoperation gestartet werden soll bzw. dass Daten auf einer Einheit des LU-Typs 2 (3274/76 oder 3790) auf einem durch den Controller zugeordneten Drucker gedruckt werden sollen. Falls diese Option nicht angegeben ist, werden die Daten an den Druckerpuffer gesendet, jedoch nicht gedruckt.

Bei Verwendung der Option PRINT können Sie sich im Abschnitt zur Option CRTL von DFHMDI über die Optionspriorität informieren.

SET(zeigerverweis)

Gibt den Zeiger an, der auf die Adresse der zugeordneten Daten gesetzt werden soll.

Der Speicherbereich, der die zugeordneten Daten enthält, besitzt dasselbe Format wie der Seitenpuffer, der bei Verwendung der Option SET im vollständigen BMS-Befehl SEND zurückgegeben wird. Weitere Anleitungen für die Verwendung der Option finden Sie unter Funktion MAPPINGDEV.

Bedingungen

Einige der folgenden Bedingungen können kombiniert auftreten. Falls mehrere Bedingungen auftreten, wird nur die erste Bedingung an das Anwendungsprogramm übergeben.

38 INVMP SZ

Tritt auf, falls die angegebene Zuordnung zu breit für das Terminal ist oder falls ein Befehl HANDLE CONDITION OVERFLOW aktiv und die angegebene Zuordnung zu lang für das durch MAPPINGDEV angegebene Terminal ist.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

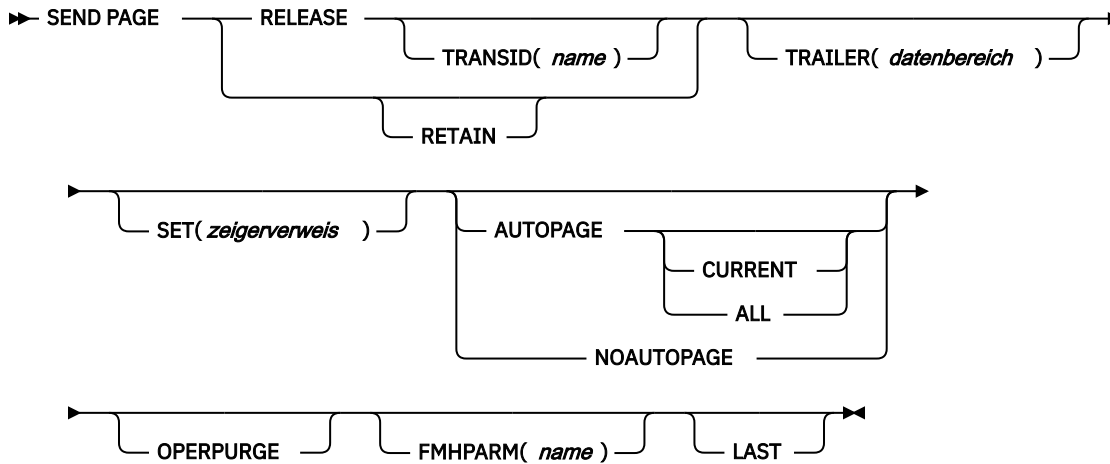
Tritt auf, falls das durch MAPPINGDEV angegebene Terminal nicht vorhanden ist, BMS nicht unterstützt oder kein 3270-Drucker bzw. keine 3270-Anzeige ist.

Standardaktion: Die Task wird abnormal beendet.

SEND PAGE

Sendet die letzte Seite der Daten.

SEND PAGE



Bedingungen: IGRQCD, INVREQ, RETPAGE, TSIOERR, WRBRK

Beschreibung

Der Befehl SEND PAGE schließt eine logische BMS-Nachricht ab. Er bewirkt, dass BMS einen einheitenabhängigen Datenstrom für die letzte (und möglicherweise einzige) Seite von Daten generiert. Diese letzte Seite ist in der Regel nur teilweise gefüllt. Der Befehl SEND PAGE steht nur bei vollständiger BMS zur Verfügung. Weitere Informationen zu BMS finden Sie unter [Basic Mapping Support](#).

Mithilfe von Optionen können Sie angeben, wie viel Steuerung der Terminalbediener über die Disposition der logischen Nachricht erhalten soll (AUTOPAGE, NOAUTOPAGE und OPERPURGE), Sie können bestimmen, ob die Steuerung nach der Übertragung der logischen Nachricht an das Anwendungsprogramm zurückgegeben werden soll (RELEASE oder RETAIN), Sie können angehängte Daten zu einer logischen Textnachricht hinzufügen (TRAILER) und Sie können den einheitenabhängigen Datenstrom für die letzte Seite einer logischen Nachricht an das Anwendungsprogramm zurückgeben lassen (SET). Falls es sich um eine Nachricht mit Seitenwechsel handelt, wird die letzte Seite der logischen Nachricht an den temporären Speicher übertragen und die Seitenwechseltransaktion für den Terminalbediener gestartet. Handelt es sich um eine logische Terminalnachricht, wird die letzte Seite an das Terminal übertragen.

Dieser Befehl wird nur bei vollständiger BMS unterstützt.

Optionen

ALL

Gibt an, dass die Übertragung der aktuellen Seite beendet werden soll und keine weiteren Seiten übertragen werden sollen, wenn bei einer 2741-Einheit die Abruftaste gedrückt wird, während eine logische BMS-Nachricht an das Terminal gesendet wird. Die logische Nachricht wird gelöscht.

AUTOPAGE

Gibt an, dass jede Seite einer logischen BMS-Nachricht an das Terminal gesendet werden soll, sobald sie verfügbar ist. Falls für das Terminal durch die RDO-Option AUTOPAGE(NO) für TYPETERM ein Seitenwechsel auf Anforderung angegeben ist, wird diese Einstellung für diese logische Nachricht durch die Option AUTOPAGE außer Kraft gesetzt.

Die Option AUTOPAGE wird bei 3270-Druckern angenommen; für 3270-Bildschirmanzeigen gilt sie nicht. Falls weder AUTOPAGE noch NOAUTOPAGE angegeben ist, verwendet das Terminal den Seitenwechselstatus, der für es in der RDO-Option AUTOPAGE von TYPETERM festgelegt ist.

CURRENT

Gibt an, dass die Übertragung der aktuellen Seite beendet und die Übertragung der nächsten Seite (sofern vorhanden) begonnen werden soll, wenn bei einer 2741-Einheit die Abruftaste gedrückt wird, während eine logische BMS-Nachricht an das Terminal gesendet wird.

FMHPARM(name)

Gibt den 1 bis 8 Zeichen langen Namen der zu verwendenden externen Zuordnung an. Diese Option gilt nur für logische 3650-Einheiten mit externer Formatierung.

LAST

Gibt an, dass es sich bei dieser Operation um die letzte Ausgabeoperation für eine Transaktion und daher um das Ende einer Klammer handelt. Bei Angabe der Option RELEASE wird die Option LAST angenommen, sofern nicht mit dem Befehl SEND PAGE eine Weiterleitungsoperation beendet wird. Diese Option gilt nur für logische Einheiten.

NOAUTOPAGE

Gibt an, dass Seiten einer logischen BMS-Nachricht jeweils nacheinander an das Terminal gesendet werden sollen. BMS sendet die erste Seite an das Terminal, sobald das Terminal verfügbar ist, oder nach Anforderung durch den Terminalbediener. Nachfolgende Seiten werden an das Terminal als Antwort auf Anforderungen durch den Terminalbediener gesendet.

Falls für das Terminal durch die RDO-Option AUTOPAGE(YES) für TYPETERM ein automatischer Seitenwechsel angegeben ist, wird diese Einstellung für diese logische Nachricht durch die Option NOAUTOPAGE außer Kraft gesetzt. Bei logischen Einheiten gilt die Option NOAUTOPAGE für alle Seiten für alle logischen Einheitscodes in der logischen Nachricht. Für 3270-Drucker gilt die Option NOAUTOPAGE nicht.

OPERPURGE

Gibt an, dass CICS die logische BMS-Nachricht nur dann löschen soll, wenn dies durch den Terminalbediener angefordert wird. Wenn die Option nicht angegeben ist, löscht CICS die Nachricht, falls der Bediener Daten eingibt, bei denen es sich nicht um einen Blätterbefehl handelt.

RELEASE

Gibt an, dass die Steuerung nach dem Befehl SEND PAGE an CICS zurückgegeben werden soll.

RETAIN

Gibt an, dass die Steuerung nach dem Befehl SEND PAGE an das Anwendungsprogramm zurückgegeben werden soll, wenn der Bediener das Anzeigen der Seiten beendet hat.

SET(zeigerverweis)

Gibt den Zeiger an, der auf die Adresse der Ausgabedaten gesetzt werden soll.

Die Option SET gibt an, dass die letzte oder einzige Seite an das Anwendungsprogramm zurückgegeben werden soll. Der Zeiger wird auf die Adresse der aktuellen Seite gesetzt. Es wird eine Liste von Adressen erstellt und bei Ausführung des Befehls ROUTE gibt es für jede Einheit einen Adresseneintrag. Falls der Befehl ROUTE nicht ausgeführt wird, enthält die Liste nur einen einzigen Eintrag.

Das Anwendungsprogramm erhält die Steuerung entweder sofort nach dem Befehl SEND PAGE zurück (falls die aktuelle Seite noch nicht fertiggestellt ist) oder aber bei dem Kennsatz, der in einem Befehl HANDLE CONDITION RETPAGE angegeben ist (falls die Seite fertiggestellt wurde).

Bei Angabe von TIOAPFX=YES in der Zuordnungsdefinition enthält der zurückgegebene Zeiger die Adresse des Präfix für den Ein-/Ausgabebereich des Terminals (Terminal Input Output Area, TIOA). Die Benutzerdaten beginnen an der relativen Position X'0C' ausgehend vom Anfang des TIOA-Präfix.

TRAILER(datenbereich)

Gibt an, dass der Textdatenbereich Trailerdaten enthält, die nur am Ende der letzten Seite platziert werden sollen. Der Trailer hat das folgende Format:

2 Byte

Binäre Länge der Daten (n)

2 Byte

Binäre Null

n Byte

Daten

Weitere Informationen enthält der Abschnitt Befehl SEND PAGE.

TRANSID(name)

Gibt die 1 bis 4 alphanumerische Zeichen lange Transaktions-ID an, die für die nächste Eingabenachricht von dem Terminal verwendet werden soll, an das die Task angehängt ist. Diese ID muss für CICS durch eine RDO-Ressourcendefinition TRANSACTION definiert sein. Die Option TRANSID ist nur bei Angabe von SEND PAGE RELEASE gültig.

Falls diese Option in einem Programm angegeben ist, das sich nicht auf der höchsten logischen Ebene befindet, wird die angegebene Transaktions-ID nur dann verwendet, wenn in einem anderen Befehl SEND PAGE (oder in einem Programmsteuerbefehl RETURN), der in einem Programm auf einer höheren logischen Ebene ausgegeben wird, keine Transaktions-ID angegeben ist.

Bedingungen

57 IGREQCD

Tritt auf, wenn versucht wird, einen Befehl SEND PAGE auszuführen, nachdem ein Datenflusssteuerungsbefehl SIGNAL mit einem Code für die Anforderung der Richtungsänderung (Request change direction, RCD) von einer logischen Einheit des LU-Typs 4 empfangen wurde.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

RESP2-Werte:

200

Der Befehl ist für ein Serverprogramm von Distributed Program Link nicht zulässig.

Tritt außerdem in jedem der folgenden Fälle auf (RESP2 nicht festgelegt):

- Die Disposition (TERMINAL, PAGING oder SET) einer logischen BMS-Nachricht wurde geändert, bevor sie durch den Befehl SEND PAGE abgeschlossen wurde.
- Textdaten werden in derselben Partition oder in demselben logischen Einheitencode wie zugeordnete Daten ausgegeben, während eine logische Nachricht von BMS aktiv ist. Falls weder Partitionen noch logische Einheitencodes verwendet werden, werden Textdaten in derselben logischen Nachricht wie zugeordnete Daten ausgegeben.
- Die Option TRAILER war angegeben, als eine logische Nachricht beendet wurde, die nur durch Befehle SEND MAP erstellt wurde.
- Während der Überlaufverarbeitung werden Daten an einen anderen logischen Einheitencode als dem Code gesendet, der den Seitenüberlauf verursacht hat.
- Die Länge des Trailers ist ein negativer Wert.

Standardaktion: Die Task wird abnormal beendet.

32 RETPAGE

Tritt auf, falls die Option SET angegeben ist und die letzte oder einzige fertiggestellte Seite für die Rückgabe an das Anwendungsprogramm bereit ist.

Standardaktion: Die Steuerung wird unmittelbar nach dem BMS-Befehl SEND PAGE an das Anwendungsprogramm zurückgegeben.

35 TSIOERR

Tritt auf, falls ein nicht behebbarer Ein-/Ausgabefehler für den temporären Speicher aufgetreten ist.

Standardaktion: Die Task wird abnormal beendet.

03 WRBRK

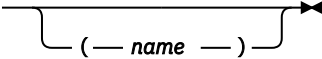
Tritt auf, falls der Befehl SEND PAGE unterbrochen wurde, weil der Terminalbediener die Abruftaste gedrückt hat. Diese Bedingung gilt nur für das Kommunikationsterminal IBM 2741 und nur dann, wenn die Unterbrechung des Schreibvorgangs für CICS unterstützt wird.

Standardaktion: Die Bedingung wird ignoriert.

SEND PARTNSET

Ordnet die in der Option PARTNSET angegebene Partitionsgruppe dem Anwendungsprogramm zu.

SEND PARTNSET

➡ SEND PARTNSET  (— *name* —)

Bedingungen: INVPARTNSET, INVREQ

Beschreibung

Der Befehl SEND PARTNSET ordnet die in der Option PARTNSET angegebene Partitionsgruppe dem Anwendungsprogramm zu. Falls der Name der Partitionsgruppe nicht angegeben ist, wird das Terminal auf seinen (nicht partitionierten) Basiszustand zurückgesetzt. Dieser Befehl ist nur bei standardmäßiger und vollständiger BMS verfügbar. Weitere Informationen zu BMS finden Sie unter [Basic Mapping Support](#).

Anmerkung: Ein Befehl PARTNSET darf nicht unmittelbar auf einen Befehl RECEIVE folgen. Zwischen den beiden Befehlen muss ein Befehl SEND MAP, SEND TEXT oder SEND CONTROL ausgegeben werden, damit die Partitionsgruppe an das Terminal gesendet wird.

Bedingungen

Die folgenden Bedingungen können kombiniert auftreten. Falls beide Bedingungen auftreten, wird nur die erste Bedingung an das Anwendungsprogramm übergeben.

64 INVPARTNSET

Tritt auf, falls die im Befehl SEND PARTNSET benannte Partitionsgruppe keine gültige Partitionsgruppe ist (sondern beispielsweise eine Maskengruppe).

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

RESP2-Werte:

200

Der Befehl ist für ein Serverprogramm von Distributed Program Link nicht zulässig.

Diese Bedingung (RESP2 nicht festgelegt) tritt auch in der folgenden Situation auf:

- Ein Befehl SEND PARTNSET wurde ausgegeben, als eine logische Nachricht aktiv war.

Standardaktion: Die Task wird abnormal beendet.

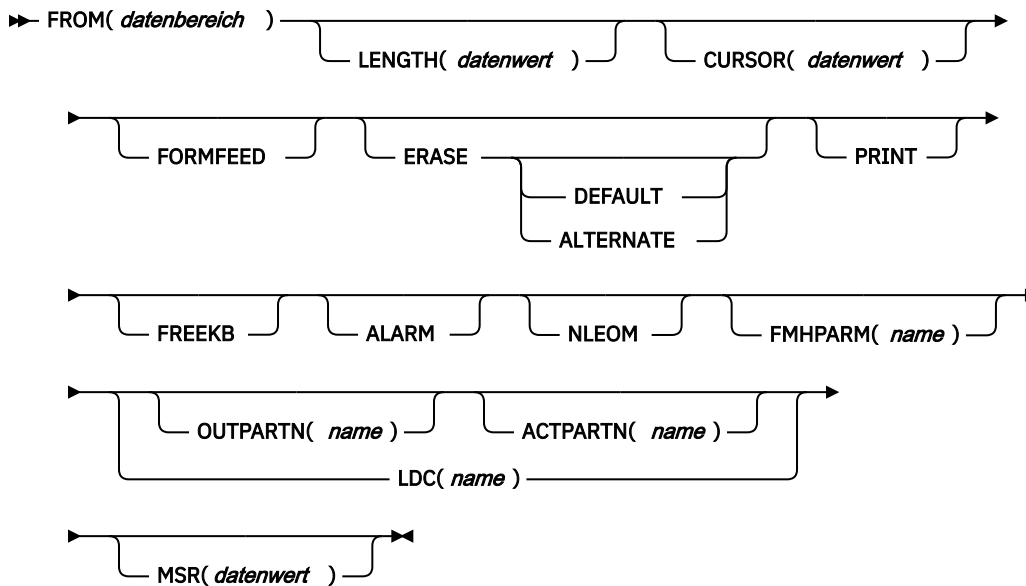
SEND TEXT

Sendet Daten ohne Zuordnung. Die Schlüsselwörter werden unterteilt in diejenigen, die von standardmäßiger und vollständiger BMS unterstützt werden. Weitere Informationen zu BMS finden Sie unter [Basic Mapping Support](#).

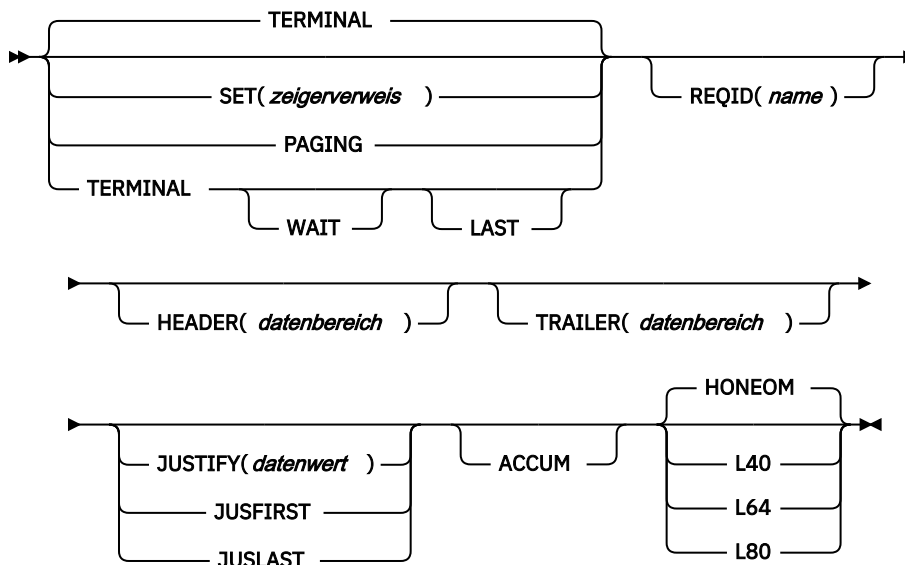
SEND TEXT

➡ SEND TEXT ➡

SEND TEXT (standardmäßige BMS)



SEND TEXT (vollständige BMS)



Bedingungen: IGRREQCD, IGRREQID, INVLDC, INVPARTN, INVREQ, LENGERR, RETPAGE, TSIOERR, WRBRK

Beschreibung

Der Befehl SEND TEXT sendet Textdaten ohne Zuordnung. Der Text wird in Zeilen mit derselben Breite wie das Terminal eingeteilt, damit es nicht zu zeilenübergreifenden Worttrennungen kommt. Falls der Text eine Seite überschreitet, wird er in Seiten unterteilt, die mit anwendungsdefinierten Headern und Trailern in das Terminal passen.

Wenn Sie den Befehl SEND TEXT mit einer der Optionen ALARM, FREEKB, FRSET, HONEOM, L40, L64, L80 oder PRINT verwenden, können Sie sich im Abschnitt zur [Option CTRL für DFHMDI](#) über die Optionspriorität informieren.

Optionen

ACCUM

Gibt an, dass dieser Befehl zu einer Reihe von Befehlen gehört, die zur Erstellung einer logischen Nachricht verwendet werden. Die logische Nachricht wird durch einen Befehl SEND PAGE abgeschlossen oder durch einen Befehl PURGE MESSAGE gelöscht.

Die Optionen HEADER, JUSFIRST, JUSLAST, JUSTIFY und TRAILER implizieren sämtlich die Option ACCUM.

ACTPARTN(name)

Gibt den 1 bis 2 Zeichen langen Namen der zu aktivierenden Partition an. Durch das Aktivieren einer Partition wird der Cursor in die angegebene Partition verschoben und die Tastatur für die angegebene Partition entsperrt.

Falls das Zielterminal Partitionen nicht unterstützt oder keine Anwendungspartitionsgruppe festgelegt ist, wird diese Option ignoriert.

ALARM

Gibt an, dass die 3270-Funktion für akustische Signale aktiviert werden soll. Bei logischen Einheiten, die Funktionsverwaltungsheader unterstützen (mit Ausnahme von interaktiven logischen Einheiten und logischen Einheiten mit Stapel) wird BMS durch die Option ALARM angewiesen, die Markierung für Alarmsignale im Funktionsverwaltungsheader festzulegen.

ALTERNATE

Legt die Verwendung der alternativen Anzeigegröße durch das Terminal fest.

CURSOR(datenwert)

Gibt die Position an, an die der 3270- oder 3604-Cursor nach Abschluss eines Befehls SEND TEXT zurückgesetzt werden soll.

Der Datenwert muss ein Halbwort-Binärwert sein, der die Cursorposition bezogen auf Null angibt; welcher Wertebereich angegeben werden kann, ist von der Größe der verwendeten Anzeige abhängig.

Diese Option setzt jede Option IC des Operanden ATTRB von DFHMDF außer Kraft. Bei Verwendung der Option ACCUM wird der Cursor anhand des zuletzt angegebenen Wertes von CURSOR positioniert.

In der Option CURSOR muss ein positiver Wert angegeben werden. Ein negativer Wert führt zu unvorhersehbaren Ergebnissen.

DEFAULT

Legt die Verwendung der Standardanzeigegröße durch das Terminal fest.

ERASE

Gibt an, dass der Anzeigendruckerpuffer oder die Partition gelöscht und der Cursor in die linke obere Ecke der Anzeige zurückgesetzt werden soll. (Diese Option gilt nur für die 3270- bzw. 8775-Anzeige und die 3604-Tastaturanzeige.)

Die erste Ausgabeoperation in einer Transaktion oder in einer Serie von Pseudodialogtransaktionen sollte immer die Option ERASE angeben. Für Transaktionen, die an 3270-Anzeigen oder -Drucker angehängt sind, stellt diese Option (sofern sie nicht explizit durch die Option DEFAULT oder ALTERNATE überschrieben wird) außerdem sicher, dass die richtige Anzeigegröße ausgewählt wird, die für die Transaktion durch die Option SCRNSIZE in der RDO-Definition PROFILE definiert ist.

FMHPARM(name)

Gibt den 1 bis 8 Zeichen langen Namen der zu verwendenden externen Zuordnung an. (Diese Option gilt nur für logische 3650-Einheiten mit externer Formatierung.)

FORMFEED

Gibt an, dass eine neue Seite erforderlich ist. Bei 3270-Druckern und -Anzeigen muss das FORMFEED-Zeichen am Anfang des Puffers positioniert sein. Das Anwendungsprogramm muss daher sicherstellen, dass diese Pufferposition nicht durch Masken- oder Textdaten überschrieben wird. Falls das Zielterminal die Option FORMFEED nicht unterstützt (also die RDO-Option FORMFEED für TYPE-TERM nicht verwendet wurde), wird diese Option ignoriert.

Die Option FORMFEED kann bei einem beliebigen Befehl SEND TEXT ACCUM angegeben werden. Sie muss innerhalb einer physischen Seite nur ein einziges Mal angegeben sein, weil sie in jedem Fall einen Seitenvorschub (FORMFEED) am Beginn der physischen Seite erzwingt. Um einen Seitenvorschub am Beginn eines bestimmten Befehls SEND TEXT ACCUM zu erzwingen, verwenden Sie stattdessen die Option JUSFIRST.

FREEKB

Gibt an, dass die 3270-Tastatur entsperrt werden soll, nachdem die Daten geschrieben wurden. Wenn die Option FREEKB nicht angegeben wird, bleibt die Tastatur gesperrt.

Bei Verwendung der Option FREEKB können Sie sich im Abschnitt zur [Option CRTL von DFHMDI](#) über die Optionspriorität informieren.

Es ist zu beachten, dass der Status für die Tastatursperre auf einem Terminal, das Partitionen unterstützt, für jede Partition separat verwaltet wird.

FROM(datenbereich)

Gibt den Datenbereich an, in dem die Daten enthalten sind, die gesendet werden sollen.

HEADER(datenbereich)

Gibt die Headerdaten an, die an den Beginn jeder Seite mit Textdaten gestellt werden sollen. Der Header hat das folgende Format:

2 Byte

Binäre Länge der Daten (n)

1 Byte

Seitennummerierung erforderlich oder nicht (leer)

1 Byte

Reserviertes Feld

n Byte

Daten

Weitere Informationen enthält der Abschnitt [Header- und Trailerformat](#).

HONEOM

Gibt an, dass die Standardzeilenlänge des Druckers verwendet werden soll. Diese Länge sollte mit der Länge, die durch die RDO-Optionen PAGESIZE oder ALTPAGE für TYPETERM angegeben ist, sowie mit der Breite der Druckerwalze identisch sein; andernfalls werden die Daten möglicherweise nicht korrekt formatiert.

Bei Verwendung der Option HONEOM können Sie sich im Abschnitt zur [Option CRTL von DFHMDI](#) über die Optionspriorität informieren.

JUSFIRST

Gibt an, dass die Textdaten am Beginn der Seite angeordnet sein sollen. Alle teilweise formatierten Seiten aus vorherigen Anforderungen werden als fertiggestellt betrachtet. Falls die Option HEADER angegeben ist, geht den Daten der Header voraus. Lesen Sie in diesem Zusammenhang auch die Beschreibung der Option JUSTIFY.

JUSLAST

Gibt an, dass die Textdaten am Ende der Seite angeordnet sein sollen. Die Seite gilt als fertiggestellt, nachdem die Anforderung verarbeitet wurde. Falls die Option TRAILER angegeben ist, folgt auf die Daten der Trailer. Lesen Sie in diesem Zusammenhang auch die Beschreibung der Option JUSTIFY.

JUSTIFY(datenwert)

Gibt die Zeile der Seite an, an der die Textdaten positioniert sein sollen. Der Datenwert muss ein Halbwort-Binärwert zwischen 1 und 240 sein. Die Sonderwerte -1 und -2 können zwar nicht als Konstanten verwendet werden, aber dynamisch zur Angabe von JUSFIRST bzw. JUSLAST genutzt werden.

LAST

Gibt an, dass es sich bei dieser Operation um die letzte Ausgabeoperation für eine Transaktion und daher um das Ende einer Klammer handelt. Diese Option gilt nur für logische Einheiten.

LDC(name)

Gibt ein zweistelliges mnemonisches Zeichen an, mit dessen Hilfe der logische Einheitencode (Logical Device Code, LDC) ermittelt wird, der im Funktionsverwaltungsheder an die logische Einheit übertragen werden soll. Das mnemonische Zeichen gibt einen LDC-Eintrag an, der durch ein Makro DFHTCT TYPE=LDC definiert wird.

Bei Angabe eines logischen Einheitencodes verwendet BMS den Einheitentyp, die Seitengröße und den Seitenstatus, die dem logischen Einheitencode zugeordnet sind, um die Nachricht zu formatieren. Diese Werte werden aus der erweiterten lokalen LDC-Tabelle für die logische Einheit entnommen, falls sie eine solche Tabelle besitzt. Für den Fall, dass die logische Einheit nur eine lokale (nicht erweiterte) LDC-Tabelle besitzt, werden die Werte aus der LDC-Tabelle des Systems entnommen. Der numerische Wert des logischen Einheitencodes wird aus der lokalen LDC-Tabelle abgerufen, es sei denn, dass dies eine nicht erweiterte Tabelle ist und der Wert nicht angegeben ist. In diesem Fall wird er aus der Systemtabelle abgerufen.

LENGTH(datenwert)

Gibt die Länge der zu sendenden Daten in Form eines Halbwort-Binärwertes an. Eine Beschreibung einer sicheren Obergrenze finden Sie unter „[LENGTH-Optionen in CICS-Befehlen](#)“ auf Seite 10.

L40, L64 oder L80

Gibt die Zeilenlänge für einen 3270-Drucker an; hierdurch wird eine Zeilenschaltung und ein Zeilenvorschub erzwungen, nachdem 40, 64 oder 80 Zeichen in einer Zeile gedruckt wurden. Wenn diese Einstellung von der Seitenbreite abweicht, die durch die RDO-Optionen PAGESIZE oder ALTPAGE für TYPETERM angegeben sind, sind mangelhafte Ergebnisse wahrscheinlich.

Bei Verwendung der Optionen können Sie sich im Abschnitt zur [Option CTRL von DFHMDI](#) über die Optionspriorität informieren.

MSR(datenwert)

Gibt den 4 Byte umfassenden Datenwert an, mit dem der 10/63-Magnetstreifenleser gesteuert wird, der an ein 8775- oder 3643-Terminal angeschlossen ist. In DFHMSRCA wird eine Reihe von Konstanten bereitgestellt, die das Festlegen dieses 4 Byte umfassenden Bereichs unterstützen. Eine vollständige Liste finden Sie unter „[Steuerwertkonstanten für Magnetcodeleser \(DFHMSRCA\)](#)“ auf Seite 845. Falls die RDO-Option MSRCONTROL für TYPETERM nicht verwendet wurde, wird diese Option ignoriert.

NLEOM

Gibt an, dass Daten für einen 3270-Drucker oder eine 3275-Anzeige mit Druckeradapterfunktion mit Leerzeichen und Zeilenvorschubzeichen erstellt werden sollen und dass am Ende der Daten ein Zeichen für das Nachrichtenende eingefügt werden soll. Beim Drucken der Daten bewirkt jedes Zeilenvorschubzeichen, dass die Druckausgabe in der nächsten Zeile fortgesetzt wird, und das Zeichen für das Nachrichtenende beendet die Druckausgabe.

Diese Option muss im ersten Befehl SEND TEXT angegeben sein, der zum Erstellen einer logischen Nachricht verwendet wird. Die Option wird ignoriert, falls die Einheit, die die Nachricht (direkt oder weitergeleitet) empfängt, keine der oben genannten Einheiten ist.

Falls diese Option verwendet wird, ist weder eine Pufferaktualisierung noch eine Attributänderung von zuvor in den Puffer geschriebenen Feldern zulässig. CICS bezieht bei jedem Schreibvorgang für das Terminal die Option ERASE ein.

Das Zeilenvorschubzeichen belegt eine Pufferposition. Eine Reihe von Pufferpositionen, deren Anzahl dem Wert der RDO-Optionen PAGESIZE oder ALTPAGE von TYPETERM für dieses Terminal entspricht, sind für Daten nicht verfügbar. Dies kann dazu führen, dass Daten im Puffer umlaufen; in einem solchen Fall muss der Wert für PAGESIZE herabgesetzt werden.

Die Option NLEOM setzt die Option ALARM außer Kraft, falls sie angegeben ist.

OUTPARTN(name)

Gibt den 1 bis 2 Zeichen langen Namen der Partition an, an die Daten gesendet werden sollen. Falls das Terminal Partitionen nicht unterstützt oder ihm keine Anwendungspartitionsgruppe zugeordnet ist, wird diese Option ignoriert. Wenn es eine Anwendungspartitionsgruppe gibt und die Option OUTPARTN nicht angegeben wird, werden die Daten an die Partition gesendet, die im Operanden PARTN

der Zuordnungsdefinition DFHMSD oder DFHMDI benannt ist. Falls keine Zuordnungen verwendet werden oder es keinen Operanden PARTN gibt, wird die Ausgabe an die erste Partition in der Partitionsgruppe gesendet.

PAGING

Gibt an, dass die Ausgabedaten nicht sofort an das Terminal gesendet, sondern im temporären Speicher abgelegt und als Antwort auf Blätterbefehle angezeigt werden sollen, die durch den Terminalbediener eingegeben werden.

Falls die Option PAGING mit einem Präfix REQID angegeben wird, das für Warteschlange für temporären Speicher definiert ist, die als wiederherstellbar definiert sind, stellt CICS die Nachrichtenwiederherstellung für logische Nachrichten bereit, falls die Task einen Synchronisationspunkt erreicht hat.

PRINT

Gibt an, dass auf einem 3270-Drucker oder einer 3275-Einheit mit Druckeradapterfunktion eine Druckoperation gestartet werden soll bzw. dass Daten auf einer Einheit des LU-Typs 2 (3274/76 oder 3790) auf einem durch den Controller zugeordneten Drucker gedruckt werden sollen. Falls diese Option nicht angegeben ist, werden die Daten an den Druckerpuffer gesendet, jedoch nicht gedruckt.

Bei Verwendung der Option PRINT können Sie sich im Abschnitt zur [Option CRTL von DFHMDI](#) über die Optionspriorität informieren.

REQID(name)

Gibt ein zweistelliges Präfix an, das als Teil einer Kennung des temporären Speichers für die CICS-Nachrichtenwiederherstellung verwendet werden soll. Für jede logische Nachricht kann nur ein einziges Präfix angegeben werden. Das Standardpräfix ist **.

Die BMS-Nachrichtenwiederherstellung steht für eine logische Nachricht nur dann zur Verfügung, wenn die Option PAGING in den BMS-Befehlen SEND angegeben ist und wenn der Synchronisationspunkt erreicht wurde.

SET(zeigerverweis)

Gibt den Zeiger an, der auf die Adresse der Daten gesetzt werden soll. Die Option SET gibt an, dass fertiggestellte Seiten an das Anwendungsprogramm zurückgegeben werden sollen. Der Zeiger wird auf die Adresse einer Liste von fertiggestellten Seiten gesetzt.

Das Anwendungsprogramm erhält die Steuerung entweder sofort nach dem BMS-Befehl SEND zurück (falls die aktuelle Seite noch nicht fertiggestellt ist) oder aber bei dem Kennsatz, der in einem Befehl HANDLE CONDITION RETPAGE angegeben ist (falls die Seite fertiggestellt wurde).

Bei Angabe von TIOAPFX=YES in der Zuordnungsdefinition enthält der zurückgegebene Zeiger die Adresse des Präfix für den Ein-/Ausgabebereich des Terminals (Terminal Input Output Area, TIOA). Die Benutzerdaten beginnen an der relativen Position X'0C' ausgehend vom Anfang des TIOA-Präfix.

TERMINAL

Gibt an, dass die Daten an das Terminal gesendet werden sollen, von dem die Transaktion stammt.

TRAILER(datenbereich)

Gibt an, dass der Textdatenbereich Trailerdaten enthält, die am Ende jeder Ausgabeseite platziert werden sollen. Der Trailer hat das folgende Format:

2 Byte

Binäre Länge der Daten (n)

1 Byte

Seitennummerierung erforderlich oder nicht (leer)

1 Byte

Reserviertes Feld

n Byte

Daten

Weitere Informationen enthält der Abschnitt [Header- und Trailerformat](#).

WAIT

Gibt an, dass die Steuerung erst nach Abschluss der Ausgabeoperation an das Anwendungsprogramm zurückgegeben werden soll.

Falls die Option WAIT nicht angegeben ist, wird die Steuerung an das Anwendungsprogramm zurückgegeben, sobald die Ausgabeoperation gestartet worden ist. Ein nachfolgender Eingabe- oder Ausgabebefehl (Terminalsteuerung, BMS oder Stapeldatenaustausch) führt dazu, dass sich das Anwendungsprogramm im Wartestatus befindet, bis der vorherige Befehl abgeschlossen wurde.

Bedingungen

57 IGREQCD

Tritt auf, wenn versucht wird, einen Befehl SEND TEXT auszuführen, nachdem ein Datenflussteueringbefehl SIGNAL mit einem Code für die Anforderung der Richtungsänderung (Request change direction, RCD) von einer logischen Einheit des LU-Typs 4 empfangen wurde.

Standardaktion: Die Task wird abnormal beendet.

39 IREQID

Tritt auf, falls das in der Option REQID in einem BMS-Befehl SEND angegebene Präfix nicht mit dem von einer vorherigen Option REQID oder mit dem standardmäßig für diese logische Nachricht erstellten Präfix, also REQID (**), identisch ist.

Standardaktion: Die Task wird abnormal beendet.

41 INVLDC

Tritt auf, falls das angegebene mnemonische Zeichen für den logischen Einheitencode (LDC) nicht in der LDC-Liste für die logische Einheit enthalten ist.

Standardaktion: Die Task wird abnormal beendet.

65 INVPARTN

Tritt auf, falls die angegebene Partition nicht in der Partitionsgruppe definiert ist, die dem Anwendungsprogramm zugeordnet ist.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

RESP2-Werte:

200

Der Befehl ist für ein Serverprogramm von Distributed Program Link nicht zulässig.

Diese Bedingung tritt außerdem in einer der folgenden Situationen auf (RESP2 nicht festgelegt):

- Textdaten werden in derselben Partition oder in demselben logischen Einheitencode wie zugeordnete Daten ausgegeben, während eine logische Nachricht von BMS aktiv ist. Falls weder Partitionen noch logische Einheitencodes verwendet werden, werden Textdaten in derselben logischen Nachricht wie zugeordnete Daten ausgegeben.
- Während der Überlaufverarbeitung werden Daten an einen anderen logischen Einheitencode als dem Code gesendet, der den Seitenüberlauf verursacht hat.
- Die Länge eines Headers in einem Befehl SEND TEXT ist ein negativer Wert.
- Die Länge eines Trailers in einem Befehl SEND TEXT ist ein negativer Wert.

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

Tritt auf, falls in der Option LENGTH ein Wert angegeben ist, der außerhalb des gültigen Bereichs liegt.

Standardaktion: Die Task wird abnormal beendet.

32 RETPAGE

Tritt auf, falls die Option SET angegeben ist und eine fertiggestellte Seite für die Rückgabe an das Anwendungsprogramm bereit ist.

Standardaktion: Die Steuerung wird unmittelbar nach dem BMS-Befehl SEND TEXT an das Anwendungsprogramm zurückgegeben.

35 TSIOERR

Tritt auf, falls ein nicht behebbarer Ein-/Ausgabefehler für den temporären Speicher aufgetreten ist.

Standardaktion: Die Task wird abnormal beendet.

03 WRBRK

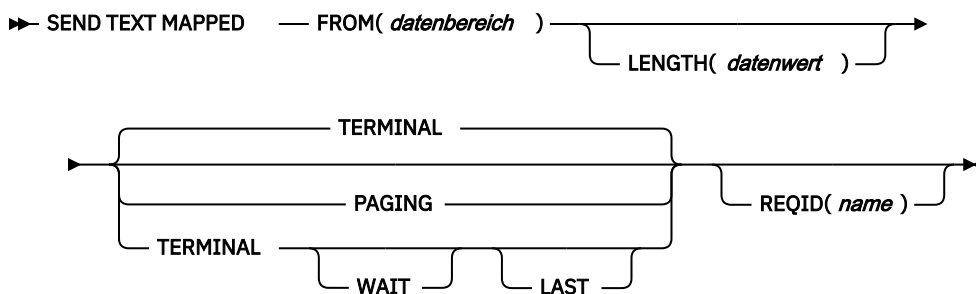
Tritt auf, falls ein Befehl SEND unterbrochen wurde, weil der Terminalbediener die Abruftaste gedrückt hat. Diese Bedingung gilt nur für das Kommunikationsterminal IBM 2741 und nur dann, wenn die Unterbrechung des Schreibvorgangs für CICS unterstützt wird.

Standardaktion: Die Bedingung wird ignoriert.

SEND TEXT MAPPED

Sendet Daten mit Zuordnung. Wird nur von vollständiger BMS bereitgestellt.

SEND TEXT MAPPED



Bedingungen: IGREQCD, IGREQID, INVREQ, TSIOERR, WRBRK

Beschreibung

Der Befehl SEND TEXT MAPPED sendet eine Seite eines einheitenabhängigen Datenstroms, die zuvor durch BMS erstellt und mit der Option SET an das Anwendungsprogramm zurückgegeben wurde.

Er darf ausschließlich zum Senden von Daten verwendet werden, die zuvor durch einen BMS-Befehl SEND unter Angabe der Option SET generiert worden sind. Er referenziert einen 4 Byte umfassenden Seitensteuerungsbereich (Page Control Area, PGA), den BMS am Ende des einheitenabhängigen Datenstroms platziert hat.

Die Länge des einheitenabhängigen Datenstroms, die im Feld TIOATDL des durch die Option SET zurückgegebenen Seitenpuffers festgelegt ist, beinhaltet nicht den PGA. Die Option LENGTH des Befehls SEND TEXT MAPPED sollte ausgehend von diesem Feld TIOATDL festgelegt sein und bezieht daher den PGA nicht ein. Falls das Anwendungsprogramm jedoch den durch die Option SET zurückgegebenen Puffer kopiert, sollten die kopierten Daten den PGA enthalten.

Dieser Befehl wird nur bei vollständiger BMS unterstützt.

Optionen

FROM(datenbereich)

Gibt den Datenbereich an, in dem die Daten enthalten sind, die gesendet werden sollen.

LAST

Gibt an, dass es sich bei dieser Operation um die letzte Ausgabeoperation für eine Transaktion und daher um das Ende einer Klammer handelt. Diese Option gilt nur für logische Einheiten.

LENGTH(datenwert)

Gibt die Länge der zu formatierenden Daten in Form eines Halbwort-Binärwertes an. Eine Beschreibung einer sicheren Obergrenze finden Sie unter „LENGTH-Optionen in CICS-Befehlen“ auf Seite 10.

PAGING

Gibt an, dass die Ausgabedaten nicht sofort an das Terminal gesendet, sondern im temporären Speicher abgelegt und als Antwort auf Blätterbefehle angezeigt werden sollen, die durch den Terminalbediener eingegeben werden.

Falls die Option PAGING mit einem Präfix REQID angegeben wird, das für Warteschlange für temporären Speicher definiert ist, die als wiederherstellbar definiert sind, stellt CICS die Nachrichtenwiederherstellung für logische Nachrichten bereit, falls die Task einen Synchronisationspunkt erreicht hat.

REQID(name)

Gibt ein zweistelliges Präfix an, das als Teil einer Kennung des temporären Speichers für die CICS-Nachrichtenwiederherstellung verwendet werden soll. Für jede logische Nachricht kann nur ein einziges Präfix angegeben werden. Das Standardpräfix ist **.

Die BMS-Nachrichtenwiederherstellung steht für eine logische Nachricht nur dann zur Verfügung, wenn die Option PAGING in den BMS-Befehlen SEND angegeben ist und wenn der Synchronisationspunkt erreicht wurde.

TERMINAL

Gibt an, dass die Eingabedaten an das Terminal gesendet werden sollen, von dem die Transaktion stammt.

WAIT

Gibt an, dass die Steuerung erst nach Abschluss der Ausgabeoperation an das Anwendungsprogramm zurückgegeben werden soll.

Falls die Option WAIT nicht angegeben ist, wird die Steuerung an das Anwendungsprogramm zurückgegeben, sobald die Ausgabeoperation gestartet worden ist. Ein nachfolgender Eingabe- oder Ausgabebefehl (Terminalsteuerung, BMS oder Stapeldatenaustausch) führt dazu, dass sich das Anwendungsprogramm im Wartestatus befindet, bis der vorherige Befehl abgeschlossen wurde.

Bedingungen**57 IGREQCD**

Tritt auf, wenn versucht wird, einen Befehl SEND TEXT auszuführen, nachdem ein Datenflusststeuerungsbefehl SIGNAL mit einem Code für die Anforderung der Richtungsänderung (Request change direction, RCD) von einer logischen Einheit des LU-Typs 4 empfangen wurde.

Standardaktion: Die Task wird abnormal beendet.

39 IGREQID

Tritt auf, falls das in der Option REQID in einem BMS-Befehl SEND angegebene Präfix nicht mit dem von einer vorherigen Option REQID oder mit dem standardmäßig für diese logische Nachricht erstellten Präfix, also REQID (**), identisch ist.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

RESP2-Werte:

200

Eine Serveranwendung für Distributed Program Link hat in der Option CONVID die Sitzung für die Funktionsverlagerung (ihre Hauptfunktion) angegeben.

Standardaktion: Die Task wird abnormal beendet.

35 TSI0ERR

Tritt auf, falls ein nicht behebbarer Ein-/Ausgabefehler für den temporären Speicher aufgetreten ist.

Standardaktion: Die Task wird abnormal beendet.

03 WRBRK

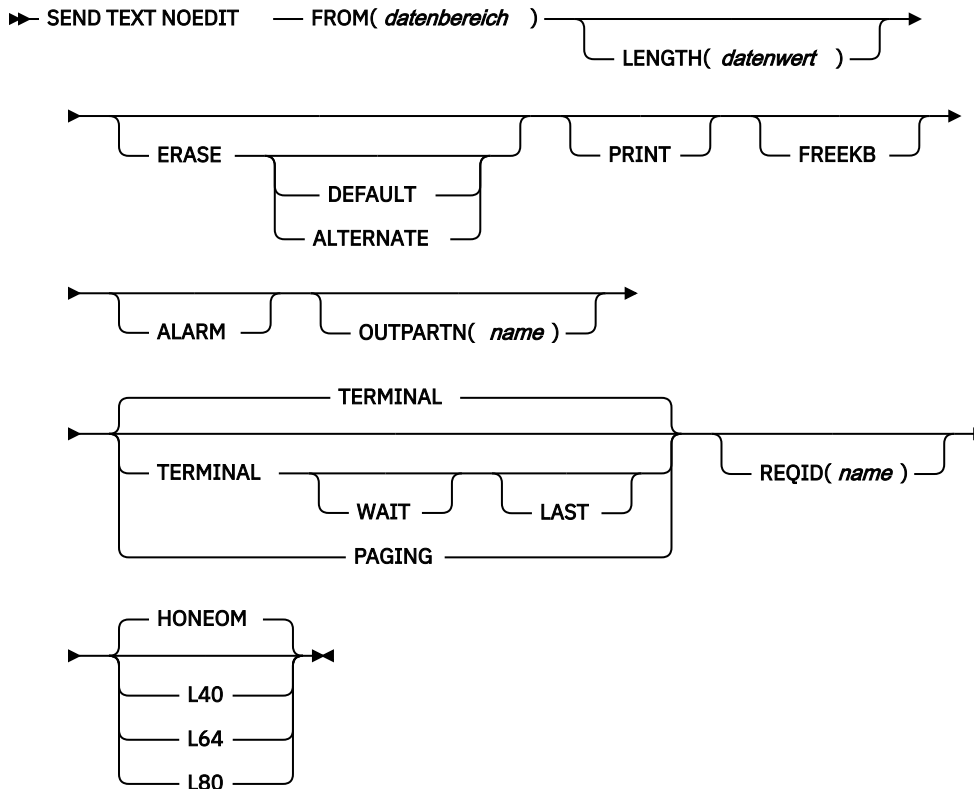
Tritt auf, falls ein Befehl SEND unterbrochen wurde, weil der Terminalbediener die Abruftaste gedrückt hat. Diese Bedingung gilt nur für das Kommunikationsterminal IBM 2741 und nur dann, wenn die Unterbrechung des Schreibvorgangs für CICS unterstützt wird.

Standardaktion: Die Bedingung wird ignoriert.

SEND TEXT NOEDIT

Sendet eine Seite. Wird nur von vollständiger BMS bereitgestellt.

SEND TEXT NOEDIT



Bedingungen: IGRQCD, IGRQID, INVREQ, INVPARTN, TSIOERR, WRBRK

Beschreibung

Der Befehl **SEND TEXT NOEDIT** sendet eine Seite eines einheitenabhängigen Datenstroms, die durch das Anwendungsprogramm erstellt wurde. Der Datenstrom kann keine strukturierten Felder enthalten. Dieser Befehl unterscheidet sich von einer Terminalsteuerung **SEND**, weil der Datenstrom in den temporären Speicher geschrieben werden und als Schnittstelle mit der Seitenwechseltransaktion des Terminalbedieners (unter Verwendung der Option **PAGING**) verbunden werden kann. Der einheitenabhängige Datenstrom kann außerdem (mit der Option **OUTPARTN**) an eine Partition gesendet werden.

Falls die Option **OUTPARTN** angegeben ist, wird der Datenstrom an die angegebene Partition gesendet. Mit diesem Befehl wird ein vom Benutzer generierter Datenstrom ausgegeben. Er unterscheidet sich dadurch von einer Terminalsteuerung **SEND**, dass Daten im temporären Speicher (mit der Option **PAGING**) ausgegeben oder wie andere BMS-Daten weitergeleitet werden können.

Wenn Sie den Befehl SEND TEXT NOEDIT mit einer der Optionen ALARM, FREEKB, FRSET, HONEOM, L40, L64, L80 oder PRINT verwenden, können Sie sich im Abschnitt zur [Option CTRL für DFHMDI](#) über die Optionspriorität informieren.

Dieser Befehl wird nur bei vollständiger BMS unterstützt.

Optionen

ALARM

Gibt an, dass die 3270-Funktion für akustische Signale aktiviert werden soll. Bei logischen Einheiten, die Funktionsverwaltungsheader unterstützen (mit Ausnahme von interaktiven logischen Einheiten und logischen Einheiten mit Stapel) wird BMS durch die Option ALARM angewiesen, die Markierung für Alarmsignale im Funktionsverwaltungsheader festzulegen.

ALTERNATE

Legt die Verwendung der alternativen Anzeigegröße durch das Terminal fest.

DEFAULT

Legt die Verwendung der Standardanzeigegröße durch das Terminal fest.

ERASE

Gibt an, dass der Anzeigendruckerpuffer oder die Partition gelöscht und der Cursor in die linke obere Ecke der Anzeige zurückgesetzt werden soll. (Diese Option gilt nur für die 3270- bzw. 8775-Anzeige und die 3604-Tastaturanzeige.)

Die erste Ausgabeoperation in einer Transaktion oder in einer Serie von Pseudodialogtransaktionen sollte immer die Option ERASE angeben. Für Transaktionen, die an 3270-Anzeigen oder -Drucker angehängt sind, stellt diese Option (sofern sie nicht explizit durch die Option DEFAULT oder ALTERNATE überschrieben wird) außerdem sicher, dass die richtige Anzeigegröße ausgewählt wird, die für die Transaktion durch die Option SCRNSIZE in der RDO-Definition PROFILE definiert ist.

FREEKB

Gibt an, dass die 3270-Tastatur entsperrt werden soll, nachdem die Daten geschrieben wurden. Wenn die Option FREEKB nicht angegeben wird, bleibt die Tastatur gesperrt.

Es ist zu beachten, dass der Status für die Tastatursperre auf einem Terminal, das Partitionen unterstützt, für jede Partition separat verwaltet wird.

Bei Verwendung der Option FREEKB können Sie sich im Abschnitt zur [Option CTRL von DFHMDI](#) über die Optionspriorität informieren.

FROM(datenbereich)

Gibt den Datenbereich an, in dem die Daten enthalten sind, die gesendet werden sollen.

HONEOM

Gibt an, dass die Standardzeilenlänge des Druckers verwendet werden soll. Diese Länge sollte mit der Länge, die durch die RDO-Optionen PAGESIZE oder ALTPAGE für TYPETERM angegeben ist, sowie mit der Breite der Druckerwalze identisch sein; andernfalls werden die Daten möglicherweise nicht korrekt formatiert.

Bei Verwendung der Option HONEOM können Sie sich im Abschnitt zur [Option CTRL von DFHMDI](#) über die Optionspriorität informieren.

LAST

Gibt an, dass es sich bei dieser Operation um die letzte Ausgabeoperation für eine Transaktion und daher um das Ende einer Klammer handelt. Diese Option gilt nur für logische Einheiten.

LENGTH(datenwert)

Gibt die Länge der zu sendenden Daten in Form eines Halbwort-Binärwertes an. Eine Beschreibung einer sicheren Obergrenze finden Sie unter [„LENGTH-Optionen in CICS-Befehlen“](#) auf Seite 10.

L40, L64 oder L80

Gibt die Zeilenlänge für einen 3270-Drucker an; hierdurch wird eine Zeilenschaltung und ein Zeilenvorschub erzwungen, nachdem 40, 64 oder 80 Zeichen in einer Zeile gedruckt wurden. Wenn diese Einstellung von der Seitenbreite abweicht, die durch die RDO-Optionen PAGESIZE oder ALTPAGE für TYPETERM angegeben sind, sind mangelhafte Ergebnisse wahrscheinlich.

Bei Verwendung der Optionen können Sie sich im Abschnitt zur [Option CTRL von DFHMDI](#) über die Optionspriorität informieren.

OUTPARTN(name)

Gibt den 1 bis 2 Zeichen langen Namen der Partition an, an die Daten gesendet werden sollen. Falls das Terminal Partitionen nicht unterstützt oder ihm keine Anwendungspartitionsgruppe zugeordnet ist, wird diese Option ignoriert. Wenn es eine Anwendungspartitionsgruppe gibt und die Option OUTPARTN nicht angegeben wird, werden die Daten an die Partition gesendet, die im Operanden PARTN der Zuordnungsdefinition DFHMSD oder DFHMDI benannt ist. Falls keine Zuordnungen verwendet werden oder es keinen Operanden PARTN gibt, wird die Ausgabe an die erste Partition in der Partitionsgruppe gesendet.

PAGING

Gibt an, dass die Ausgabedaten nicht sofort an das Terminal gesendet, sondern im temporären Speicher abgelegt und als Antwort auf Blätterbefehle angezeigt werden sollen, die durch den Terminalbediener eingegeben werden.

Falls die Option PAGING mit einem Präfix REQID angegeben wird, das für Warteschlange für temporären Speicher definiert ist, die als wiederherstellbar definiert sind, stellt CICS die Nachrichtenwiederherstellung für logische Nachrichten bereit, falls die Task einen Synchronisationspunkt erreicht hat.

PRINT

Gibt an, dass auf einem 3270-Drucker oder einer 3275-Einheit mit Druckeradapterfunktion eine Druckoperation gestartet werden soll bzw. dass Daten auf einer Einheit des LU-Typs 2 (3274/76 oder 3790) auf einem durch den Controller zugeordneten Drucker gedruckt werden sollen. Falls diese Option nicht angegeben ist, werden die Daten an den Druckerpuffer gesendet, jedoch nicht gedruckt.

Bei Verwendung der Option PRINT können Sie sich im Abschnitt zur [Option CTRL von DFHMDI](#) über die Optionspriorität informieren.

REQID(name)

Gibt ein zweistelliges Präfix an, das als Teil einer Kennung des temporären Speichers für die CICS-Nachrichtenwiederherstellung verwendet werden soll. Für jede logische Nachricht kann nur ein einziges Präfix angegeben werden. Das Standardpräfix ist **.

TERMINAL

Gibt an, dass die Daten an das Terminal gesendet werden sollen, von dem die Transaktion stammt.

WAIT

Gibt an, dass die Steuerung erst nach Abschluss der Ausgabeoperation an das Anwendungsprogramm zurückgegeben werden soll.

Falls die Option WAIT nicht angegeben ist, wird die Steuerung an das Anwendungsprogramm zurückgegeben, sobald die Ausgabeoperation gestartet worden ist. Ein nachfolgender Eingabe- oder Ausgabebefehl (Terminalsteuerung, BMS oder Stapeldatenaustausch) führt dazu, dass sich das Anwendungsprogramm im Wartestatus befindet, bis der vorherige Befehl abgeschlossen wurde.

Bedingungen

57 IGREQCD

Tritt auf, wenn versucht wird, einen Befehl SEND TEXT auszuführen, nachdem ein Datenflusststeuerungsbefehl SIGNAL mit einem Code für die Anforderung der Richtungsänderung (Request change direction, RCD) von einer logischen Einheit des LU-Typs 4 empfangen wurde.

Standardaktion: Die Task wird abnormal beendet.

39 IGREQID

Tritt auf, falls das in der Option REQID in einem BMS-Befehl SEND angegebene Präfix nicht mit dem von einer vorherigen Option REQID oder mit dem standardmäßig für diese logische Nachricht erstellten Präfix, also REQID (**), identisch ist.

Standardaktion: Die Task wird abnormal beendet.

65 INVPARTN

Tritt auf, falls die angegebene Partition nicht in der Partitionsgruppe definiert ist, die dem Anwendungsprogramm zugeordnet ist.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

RESP2-Werte:

200

Der Befehl ist für ein Serverprogramm von Distributed Program Link nicht zulässig.

Diese Bedingung tritt außerdem in einer der folgenden Situationen auf (RESP2 nicht festgelegt):

- Textdaten werden in derselben Partition oder in demselben logischen Einheitencode wie zugeordnete Daten ausgegeben, während eine logische Nachricht von BMS aktiv ist. Falls weder Partitionen noch logische Einheitencodes verwendet werden, werden Textdaten in derselben logischen Nachricht wie zugeordnete Daten ausgegeben.
- Während der Überlaufverarbeitung werden Daten an einen anderen logischen Einheitencode als dem Code gesendet, der den Seitenüberlauf verursacht hat.
- Die Länge eines Headers in einem Befehl SEND TEXT ist ein negativer Wert.
- Die Länge eines Trailers in einem Befehl SEND TEXT ist ein negativer Wert.

Standardaktion: Die Task wird abnormal beendet.

35 TSIOERR

Tritt auf, falls ein nicht behebbarer Ein-/Ausgabefehler für den temporären Speicher aufgetreten ist.

Standardaktion: Die Task wird abnormal beendet.

03 WRBRK

Tritt auf, falls ein Befehl SEND unterbrochen wurde, weil der Terminalbediener die Abruftaste gedrückt hat. Diese Bedingung gilt nur für das Kommunikationsterminal IBM 2741 und nur dann, wenn die Unterbrechung des Schreibvorgangs für CICS unterstützt wird.

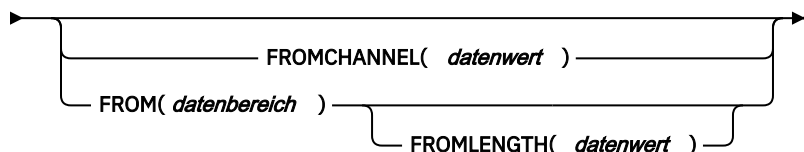
Standardaktion: Die Bedingung wird ignoriert.

SIGNAL EVENT

Gibt eine Position in einem Anwendungsprogramm an, an der ein oder mehrere Ereignisse ausgegeben werden können.

SIGNAL EVENT

►► SIGNAL EVENT(*datenwert*) →►



Bedingungen: CHANNELERR, EVENTERR, LENGERR

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl SIGNAL EVENT kennzeichnet eine Stelle in einem Anwendungsprogramm, an der eines oder mehrere Ereignisse ausgegeben werden könnten. Ereignisse werden ausgegeben, wenn die folgenden Bedingungen erfüllt sind:

- Die Ereignisverarbeitung ist aktiv.

- Es ist mindestens eine übereinstimmende Erfassungsspezifikation aktiviert. Eine Erfassungsspezifikation stimmt überein, wenn sie einen Erfassungspunkt **SIGNAL EVENT** besitzt und alle ihre Prädikate mit dem Ergebnis 'true' ausgewertet werden.

Der Befehl **SIGNAL EVENT** besitzt das primäre Prädikat **EVENT** und lässt sekundäre Prädikate für den mit **FROM** angegebenen Datenbereich oder den mit der Option **FROMCHANNEL** angegebenen Kanal und dessen Container zu. Die Daten in einem CICS-Ereignis werden infolge der Definition von **SIGNAL EVENT** in dem Geschäftsereignis ausgegeben, das die übereinstimmende Erfassungsspezifikation enthält.

Optionen

EVENT(datenwert)

Gibt eine ID (1 bis 32 Zeichen) an, die diesen Befehl **SIGNAL EVENT** kennzeichnet.

Zulässige Zeichen: A-Z a-z 0-9 \$ @ # / % & ? ! : | " = , ; < > . - und _ . Führende und eingebettete Leerzeichen sind nicht zulässig. Falls der angegebene Name weniger als 32 Zeichen umfasst, wird er mit nachgestellten Leerzeichen auf 32 Zeichen aufgefüllt.

Ereignis-IDs werden immer in EBCDIC angegeben. Die oben angegebene Gruppe der zulässigen Zeichen für Ereignis-IDs enthält einige Zeichen, die nicht in allen EBCDIC-Codepages dieselbe Darstellung besitzen. Es wird daher empfohlen, bei der Benennung von Ereignis-IDs ausschließlich die Zeichen A-Z a-z 0-9 & : = , ; < > . - und _ zu verwenden.

FROMCHANNEL(datenwert)

Gibt den 1 bis 16 Zeichen langen Namen eines Kanals an, der Daten für dieses Ereignis enthält. Sie können den Kanalnamen DFHTRANSACTION angeben, um den Transaktionskanal zu verwenden.

FROM(datenbereich)

Gibt einen Datenbereich an, der Daten für dieses Ereignis enthält.

FROMLENGTH(datenwert)

Gibt einen Vollwort-Binärwert an, der die Länge der Byte in dem mit der Option **FROM** angegebenen Datenbereich ist.

Bedingungen

122 CHANNELERR

RESP2-Werte:

2

Der in der Option **FROMCHANNEL** angegebene Kanal wurde nicht gefunden.

111 EVENTERR

RESP2-Werte:

6

Die in der Option **EVENT** angegebene ID enthält ein falsches Zeichen oder eine falsche Zeichenkombination.

22 LENGERR

RESP2-Werte:

3

Die in der Option **FROMLENGTH** angegebene Länge ist nicht größer als Null.

SIGNOFF

Führt die Abmeldung von einem Terminal aus.

SIGNOFF

➡ SIGNOFF ➡

Bedingung: INVREQ

Dieser Befehl ist threadsicher.

Beschreibung

Mit dem Befehl SIGNOFF können Sie sich bei dem Terminal oder der Hauptfunktion abmelden, bei dem/der Sie sich zuvor angemeldet haben. Nach erfolgter Abmeldung wird das Terminal auf die Sicherheitsfunktionen und Bedienermerkmale zurückgesetzt, die dem Standardbenutzer für diese CICS-Region zugeordnet sind. Die Landessprache wird auf die Landessprache des Standardbenutzers (sofern definiert) oder auf diejenige Landessprache zurückgesetzt, die der Definition des Terminals zugeordnet ist.

Wenn dieser Befehl ausgeführt wird, erkennt CICS die Abmeldung sofort und stellt die Standardattribute für das Terminal her. Die Transaktion (und alle zugehörigen taskbezogenen Benutzerexits, Funktionsverlagerungen oder verteilten Transaktionsverarbeitungen) können andere Ressourcenmanager (RMs) aufgerufen haben, beispielsweise IMS, Db2 oder VSAM. **Ob diese anderen Ressourcenmanager die Abmeldung erkennen, bevor die Transaktion beendet ist, kann nicht vorhergesehen werden.**

Die Standardattribute gelten für alle Ressourcenmanager, die durch Folgetransaktionen am Terminal aufgerufen werden.

Bedingungen

16 INVREQ

RESP2-Werte:

1

Es ist gegenwärtig kein Benutzer angemeldet. Dieser Wert könnte darauf zurückzuführen sein, dass der externe Sicherheitsmanager (ESM) von CICS nicht initialisiert wurde.

2

Es gibt kein Terminal mit dieser Task.

3

Für das Terminal dieser Task ist die Sicherheit voreingestellt.

4

Die Abmeldung wurde unter Verwendung des Transaktionsroutings versucht, ohne dass die Transaktion CRTE verwendet wurde.

18

Die CICS-ESM-Schnittstelle ist nicht initialisiert.

200

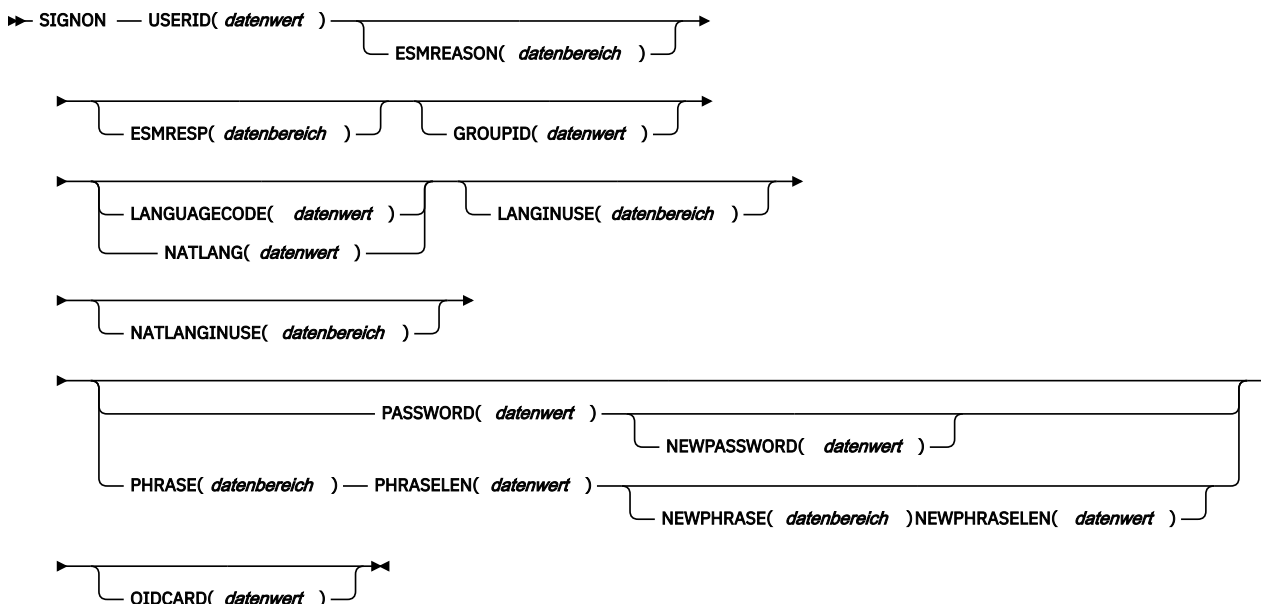
Der Befehl ist für ein Serverprogramm von Distributed Program Link nicht zulässig.

Standardaktion: Die Task wird abnormal beendet.

SIGNON

Führt die Anmeldung an einem Terminal aus.

SIGNON



Bedingungen: INVREQ, LENGERR, NOTAUTH, USERIDERR

Dieser Befehl ist threadsicher.

Beschreibung

Mit dem Befehl **SIGNON** kann Ihre Anwendung dem aktuellen Terminal eine neue Benutzer-ID zuordnen. Für die Verwendung des Befehls **SIGNON** gelten die folgenden Regeln:

- Die Anmeldeoperation ist rein terminalbezogen. Falls die Transaktion als Hauptfunktion kein Terminal besitzt, hat die Anmeldung keine Bedeutung.
- Wenn Sie einen EXEC CICS-Befehl **SIGNON** ausgeben, ändert CICS den Status des Terminals, bei dem es sich um die Hauptfunktion der Transaktion handelt, von der der Befehl ausgegeben wird.
- Die Anmeldung hat keinen Einfluss auf die Benutzer-ID und die Sicherheitsfunktionen, die gegenwärtig für die Transaktion wirksam sind, von der der Befehl ausgegeben wird. Dies hat die folgenden Gründe:
 - Die Benutzer-ID und die Sicherheitsfunktionen einer Transaktion werden zu dem Zeitpunkt eingerichtet, an dem die Transaktion angehängt wird. Während des Lebenszyklus der Transaktion können sie anschließend nicht mehr geändert werden.
 - Alle durch eine Transaktion ausgeführten Aktionen (entweder für eine lokale oder ferne Ressource bzw. für ein verbundenes System) finden in demjenigen Sicherheitskontext statt, der beim Anhängen der Transaktion eingerichtet wurde.
- Falls eine Berechtigung erforderlich ist, können Sie sich entweder mit einem gültigen Kennwort oder mit einer gültigen Kennwortphrase anmelden. Sie können jedoch weder unter Verwendung eines Kennworts für die Authentifizierung eine neue Kennwortphrase festlegen noch unter Verwendung einer Kennwortphrase für die Authentifizierung ein neues Kennwort festlegen.

Falls Sie bei der Anforderung **SIGNON** ein falsches Kennwort angeben, wird der Zähler für ungültige Versuche für die Benutzer-ID erhöht. Wenn Sie bei aufeinanderfolgenden Anforderungen **SIGNON** mehrere falsche Kennwörter angeben, wird die Benutzer-ID möglicherweise durch den *externen Sicherheitsmanager (ESM)* entzogen. CICS gibt die Nachricht DFHXS1201 aus, wenn Sie bei einer Anforderung **SIGNON** ein falsches Kennwort angeben. Wenn Sie nach einem oder mehreren ungültigen Versuchen ein richtiges Kennwort angeben, gibt CICS die Nachricht DFHXS1206 aus, die eine Anzahl der ungültigen Versuche enthält.

Der Befehl **SIGNON** schließt keine Abmeldung ein. Falls Ihr Anwendungsprogramm versucht, einem Terminal eine neue Benutzer-ID zuzuordnen, das bereits über eine angemeldete Benutzer-ID verfügt, gibt

CICS die Fehlerantwort INVREQ (RESP2-Wert: 9) zurück. Bitte beachten Sie, dass es keinen Standardwert für die Option USERID gibt.

Die Option PASSWORD wird als Parameter verwendet; dies bedeutet, dass das Kennwort möglicherweise sichtbar ist, wenn CICS über einen Hauptspeicherauszug verfügt. Daher sollten Sie den Inhalt des Feldes PASSWORD so bald wie möglich löschen, nachdem Sie es in einem Befehl SIGNON verwendet haben.

Weitere Informationen zur Verwendung der Optionen USERID und GROUPID durch CICS enthält der Abschnitt CICS-Benutzer überprüfen.

Optionen

Falls ein optionales Eingabefeld ausschließlich Leerzeichen enthält, wird es ignoriert.

ESMREASON(datenbereich)

Gibt in einem Vollwort-Binärfeld den Ursachencode zurück, den CICS vom externen Sicherheitsmanager empfängt.

Falls RACF als externer Sicherheitsmanager verwendet wird, enthält dieses Feld den RACF-Ursachencode.

ESMRESP(datenbereich)

Gibt in einem Vollwort-Binärfeld den Antwortcode zurück, den CICS vom externen Sicherheitsmanager empfängt.

Falls RACF als externer Sicherheitsmanager verwendet wird, enthält dieses Feld den RACF-Antwortcode.

GROUPID(datenwert)

Ordnet den Benutzer, der sich anmeldet, einer RACF-Benutzergruppe zu. Dies überschreibt (nur für diese Sitzung) den Standardgruppennamen, der für den Benutzer in der RACF-Datenbank angegeben ist.

LANGUAGECODE(datenwert)

Gibt die Landessprache an, die CICS auf Wunsch des angemeldeten Benutzers verwenden soll. Zur Angabe der Sprache verwenden Sie einen dreistelligen IBM Standardcode. Dies ist eine Alternative zu dem einstelligen Code, den Sie in der Option NATLANG angeben.

Mögliche Werte für die Codes finden Sie unter Codes für Landessprachen.

Anmerkung: CICS-Nachrichten werden nur in britischem Englisch, vereinfachtem Chinesisch und Japanisch unterstützt. Falls eine andere als eine dieser drei Sprachen angegeben wird, wird standardmäßig Englisch verwendet.

LANGINUSE(datenbereich)

Ermöglicht einem Anwendungsprogramm den Empfang der Landessprache, die durch den Anmeldeprozess ausgewählt wurde. Die Sprache wird durch einen dreistelligen IBM Standardcode und nicht durch den von der Option NATLANGINUSE verwendeten einstelligen Code angegeben. Dies ist eine Alternative zur vorhandenen Option NATLANGINUSE.

Mögliche Werte für die Codes finden Sie unter Codes für Landessprachen.

NATLANG(datenwert)

Gibt einen einstelligen Code für die Landessprache an, die der Benutzer während der angemeldeten Sitzung verwenden möchte.

Mögliche Werte des Codes finden Sie unter Codes für Landessprachen.

Anmerkung: CICS-Nachrichten werden nur in britischem Englisch, vereinfachtem Chinesisch und Japanisch unterstützt. Falls eine andere als eine dieser drei Sprachen angegeben wird, wird standardmäßig Englisch verwendet.

NATLANGINUSE(datenbereich)

Gibt einen einstelligen Code für die Landessprache an, die während der angemeldeten Sitzung verwendet wird. Die aktuelle Implementierung gibt immer das Zeichen E (für amerikanisches Englisch)

zurück, das der in der Option NATLANG angegebenen Sprache entspricht. Die Option NATLANGINUSE entspricht Folgendem (in absteigender Priorität):

- Der Sprache, die in der Option NATLANG des Befehls **SIGNON** angegeben ist.
- Der Sprache, die dem Benutzer zugeordnet ist. Diese Zuordnung ist im LANGUAGE-Segment des externen Sicherheitsmanagers angegeben.
- Die Sprache, die der Definition des Terminals zugeordnet ist.
- Die Sprache, die der Standard-Benutzer-ID für die CICS-Region zugeordnet ist.
- Die Standardsprache, die in den Systeminitialisierungsparametern angegeben ist.

Mögliche Werte des Codes finden Sie unter [Codes für Landessprachen](#).

NEWPASSWORD(datenwert)

Gibt ein optionales 8 Byte umfassendes Feld für das Definieren eines neuen Kennworts an. Diese Option ist nur dann gültig, wenn die Option PASSWORD ebenfalls angegeben ist. Die Eingabe einer Kennwortphrase in diesem Feld ist nicht möglich.

Wenn dieses Feld leer ist, wird es ignoriert.

Falls der externe Sicherheitsmanager Kennwörter in Groß-/Kleinschreibung nicht zulässt, wird das Kennwort in Großbuchstaben konvertiert.

NEWPHRASE(datenbereich)

Gibt ein optionales 1 bis 8 Zeichen langes neues Kennwort oder eine optionale 9 bis 100 Zeichen lange Kennwortphrase an, das/die durch den externen Sicherheitsmanager erforderlich gemacht wird. Diese Option ist nur dann gültig, wenn die Option PHRASE ebenfalls angegeben ist.

Wenn dieses Feld leer ist, wird es ignoriert.

Falls der externe Sicherheitsmanager Kennwörter in Groß-/Kleinschreibung nicht zulässt, wird das 1 bis 8 Zeichen lange Kennwort in Großbuchstaben konvertiert.

NEWPHRASELEN(datenwert)

Gibt die Länge der neuen Kennwortphrase in Form eines Vollwort-Binärwertes an. Diese Option ist nur dann gültig, wenn die Option NEWPHRASE ebenfalls angegeben ist.

Falls der Wert für NEWPHRASELEN 0 ist, wird die Option NEWPHRASE ignoriert.

OIDCARD(datenwert)

Gibt ein optionales 65 Byte umfassendes Feld an, das weitere Sicherheitsdaten aus einem Magnetstreifenleser bei 32xx-Einheiten enthält.

PASSWORD(datenwert)

Gibt ein 8 Byte langes Kennwort an, das durch den externen Sicherheitsmanager erforderlich gemacht wird.

Falls der externe Sicherheitsmanager Kennwörter in Groß-/Kleinschreibung nicht zulässt, wird das Kennwort in Großbuchstaben konvertiert.

PHRASE(datenbereich)

Gibt ein optionales 1 bis 8 Zeichen langes Kennwort oder eine optionale 9 bis 100 Zeichen lange Kennwortphrase an, das/die durch den externen Sicherheitsmanager erforderlich gemacht wird.

Falls der externe Sicherheitsmanager Kennwörter in Groß-/Kleinschreibung nicht zulässt, wird das 1 bis 8 Zeichen lange Kennwort in Großbuchstaben konvertiert.

PHRASELEN(datenwert)

Gibt die Länge der Kennwortphrase in Form eines Vollwort-Binärwertes an. Diese Option ist nur dann gültig, wenn die Option PHRASE ebenfalls angegeben ist.

USERID(datenwert)

Gibt die 8 Byte lange Benutzer-ID für die Anmeldung an.

Die angegebene Benutzer-ID wird in Großbuchstaben konvertiert.

Bedingungen

16 INVREQ

RESP2-Werte:

2

Ein Kennwort kann nicht in eine Kennwortphrase geändert werden bzw. eine Kennwortphrase nicht in ein Kennwort.

9

Das Terminal ist bereits angemeldet.

10

Dieser Task ist kein Terminal zugeordnet.

11

Für das Terminal dieser Task ist die Sicherheit voreingestellt.

12

Die Antwort von den CICS-Sicherheitsmodulen wurde nicht erkannt.

13

Der vom externen Sicherheitsmanager zurückgegebene Wert in der Option ESMRESP ist nicht von CICS klassifiziert. Eine Erläuterung der Optionen ESMRESP und ESMREASON finden Sie in der ESM-Dokumentation.

14

Die erforderliche Landessprache ist nicht verfügbar.

15

Die Anmeldung wurde unter Verwendung des Transaktionsroutings versucht, ohne dass die Transaktion CRTE verwendet wurde.

18

Die CICS ESM-Schnittstelle wurde nicht initialisiert (als Systeminitialisierungsparameter war SEC=NO angegeben).

25

Das Terminal hat einen ungültigen Typ.

26

Während der Überprüfung von SNSCOPE trat ein Fehler auf. Der Grenzwert für MVS ENQ-Anforderungen wurde erreicht.

27

Der externe Sicherheitsmanager ist nicht aktiv.

28

Die erforderliche Landessprache ist ungültig.

29

Der Benutzer ist bereits angemeldet. Dieser Code bezieht sich auf die Überprüfung der Befehlsberechtigung für die Anmeldung.

200

Der Befehl ist für ein Serverprogramm von Distributed Program Link nicht zulässig.

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

RESP2-Werte:

1

Der Wert für PHRASELEN lag außerhalb des gültigen Bereichs.

2

Der Wert für NEWPHRASELEN lag außerhalb des gültigen Bereichs.

70 NOTAUTH

RESP2-Werte:

- 1** Ein Kennwort oder eine Kennwortphrase ist erforderlich.
 - 2** Das angegebene Kennwort bzw. die angegebene Kennwortphrase ist falsch.
 - 3** Ein neues Kennwort bzw. eine neue Kennwortphrase ist erforderlich.
 - 4** Das neue Kennwort bzw. die neue Kennwortphrase ist nicht zulässig.
 - 5** Ein Wert für OI DCARD ist erforderlich.
 - 6** Der angegebene Wert für OI DCARD ist falsch.
 - 16** Die Benutzer-ID ist nicht für die Verwendung des Terminals berechtigt.
 - 17** Die Benutzer-ID ist nicht für die Verwendung der Anwendung berechtigt.
 - 19** Die Benutzer-ID wurde entzogen.
 - 20** Der Zugriff der Benutzer-ID auf die angegebene Gruppe wurde entzogen.
 - 21** Die Anmeldung ist während der Überprüfung von SECLABEL fehlgeschlagen.
 - 22** Die Anmeldung ist fehlgeschlagen, weil der externe Sicherheitsmanager gegenwärtig keine Anmeldung akzeptiert.
 - 23** Die Gruppen-ID ist dem externen Sicherheitsmanager nicht bekannt.
 - 24** Die Benutzer-ID gehört nicht zur Gruppen-ID.
- Standardaktion: Die Task wird abnormal beendet.

69 USERIDERR

RESP2-Werte:

- 8** Die Benutzer-ID ist dem externen Sicherheitsmanager nicht bekannt.
- 30** Die Benutzer-ID besteht ausschließlich aus Leerzeichen oder Nullen.

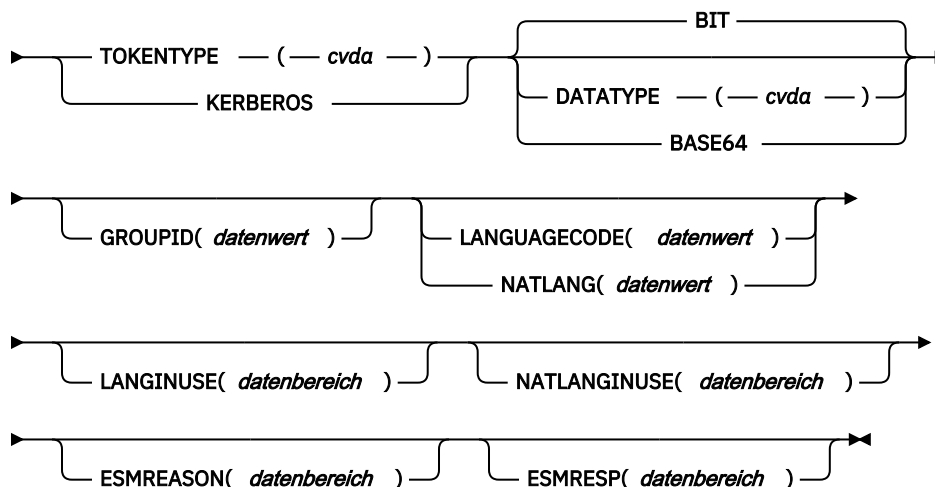
Standardaktion: Die Task wird abnormal beendet.

SIGNON TOKEN

Prüft ein Sicherheitstoken und führt mit der aus dem Token extrahierten Benutzer-ID die Anmeldung an einem Terminal aus.

SIGNON TOKEN

➔ SIGNON TOKEN(*datenbereich*) — TOKENLEN(*datenwert*) ➔



Bedingungen: INVREQ, LENGERR, NOTAUTH

Dieser Befehl ist threadsicher.

Beschreibung

Mit dem Befehl SIGNON TOKEN kann Ihre Anwendung ein Kerberos-Sicherheitstoken wie durch einen externen Sicherheitsmanager vorgegeben validieren und dem aktuellen Terminal eine neue Benutzer-ID zuordnen. Für die Verwendung des Befehls SIGNON TOKEN gelten die folgenden Regeln:

- Die Anmeldeoperation ist rein terminalbezogen. Falls die Transaktion als Hauptfunktion kein Terminal besitzt, hat die Anmeldung keine Bedeutung.
- Wenn Sie einen EXEC CICS-Befehl **SIGNON TOKEN** ausgeben, ändert CICS den Status des Terminals, bei dem es sich um die Hauptfunktion der Transaktion handelt, von der der Befehl ausgegeben wird.
- Die Anmeldung hat keinen Einfluss auf die Benutzer-ID und die Sicherheitsfunktionen, die gegenwärtig für die Transaktion wirksam sind, von der der Befehl ausgegeben wird. Dies hat die folgenden Gründe:
 - Die Benutzer-ID und die Sicherheitsfunktionen einer Transaktion werden zu dem Zeitpunkt eingerichtet, an dem die Transaktion angehängt wird. Während des Lebenszyklus der Transaktion können sie anschließend nicht mehr geändert werden.
 - Alle durch eine Transaktion ausgeführten Aktionen (entweder für eine lokale oder ferne Ressource bzw. für ein verbundenes System) finden in demjenigen Sicherheitskontext statt, der beim Anhängen der Transaktion eingerichtet wurde.
- Der Befehl SIGNON TOKEN verifiziert unter Verwendung von z/OS Security Server, dass es sich bei dem Token um ein gültiges Kerberos-Token handelt und dass das Token durch die CICS-Region verwendet werden kann.
- Falls RACF als externer Sicherheitsmanager verwendet wird, muss die CICS-Region, in der der Befehl ausgeführt wird, durch RACF autorisiert sein, damit der Kerberos-Prinzipal für das Token angefordert werden kann. Weitere Informationen finden Sie unter [RACF für Kerberos konfigurieren](#).
- Der Befehl **SIGNON TOKEN** schließt keine Abmeldung ein. Falls Ihr Anwendungsprogramm versucht, einem Terminal eine neue Benutzer-ID zuzuordnen, das bereits über eine angemeldete Benutzer-ID verfügt, gibt CICS die Fehlerantwort INVREQ (RESP2-Wert: 9) zurück.

Weitere Informationen zur Verwendung der Optionen USERID und GROUPID durch CICS enthält der Abschnitt [CICS-Benutzer überprüfen](#).

Zusätzliche Angaben über einen Sicherheitsfehler bei diesem Befehl können Sie den Fehlermeldungen entnehmen, die an das CICS-Ziel geschrieben werden.

Optionen

Falls ein optionales Eingabefeld ausschließlich Leerzeichen enthält, wird es ignoriert.

DATATYPE

Gibt den Typ der Daten im Token an. Die CVDA-Werte lauten wie folgt:

BIT

Das Token enthält Bitdaten. Dies ist der Standardwert.

BASE64

Das Token enthält Zeichendaten in Base64-Codierung. Zulässige Zeichen sind A-Z a-z 0-9 + / =

Falls Ihre Zeichendaten nicht in einer CCSID enthalten sind, die mit US EBCDIC kompatibel ist, müssen Sie sie konvertieren. Zur Konvertierung können Sie die API CONTAINER verwenden.

ESMREASON(*datenbereich*)

Gibt in einem Vollwort-Binärfeld den Ursachencode zurück, den CICS vom externen Sicherheitsmanager empfängt.

Falls RACF als externer Sicherheitsmanager verwendet wird, enthält dieses Feld den RACF-Ursachencode.

Der externe Sicherheitsmanager gibt nicht immer Antwort- und Ursachencodes an CICS zurück. Überprüfen Sie neben den Werten von ESMRESP und ESMREASON unbedingt auch die Werte von EIBRESP und EIBRESP2, die durch diesen Befehl zurückgegeben werden.

ESMRESP(*datenbereich*)

Gibt in einem Vollwort-Binärfeld den Antwortcode zurück, den CICS vom externen Sicherheitsmanager empfängt.

Falls RACF als externer Sicherheitsmanager verwendet wird, enthält dieses Feld den RACF-Antwortcode.

Der externe Sicherheitsmanager gibt nicht immer Antwort- und Ursachencodes an CICS zurück. Überprüfen Sie neben den Werten von ESMRESP und ESMREASON unbedingt auch die Werte von EIBRESP und EIBRESP2, die durch diesen Befehl zurückgegeben werden.

GROUPID(*datenwert*)

Ordnet den Benutzer, der sich anmeldet, einer RACF-Benutzergruppe zu. Dies überschreibt (nur für diese Sitzung) den Standardgruppennamen, der für den Benutzer in der RACF-Datenbank angegeben ist.

LANGUAGECODE(*datenwert*)

Gibt die Landessprache an, die CICS auf Wunsch des angemeldeten Benutzers verwenden soll. Zur Angabe der Sprache verwenden Sie einen dreistelligen IBM Standardcode. Dies ist eine Alternative zu dem einstelligen Code, den Sie in der Option NATLANG angeben.

Mögliche Werte für die Codes finden Sie in [Codes für Landessprachen](#).

Anmerkung: CICS-Nachrichten werden nur in britischem Englisch, vereinfachtem Chinesisch und Japanisch unterstützt. Falls eine andere als eine dieser drei Sprachen angegeben wird, wird standardmäßig Englisch verwendet.

LANGINUSE(*datenbereich*)

Ermöglicht einem Anwendungsprogramm den Empfang der Landessprache, die durch den Anmeldeprozess ausgewählt wurde. Die Sprache wird durch einen dreistelligen IBM Standardcode und nicht durch den von der Option NATLANGINUSE verwendeten einstelligen Code angegeben. Dies ist eine Alternative zur vorhandenen Option NATLANGINUSE.

Mögliche Werte für die Codes finden Sie unter [Codes für Landessprachen](#).

NATLANG(*datenwert*)

Gibt ein einstelliges Feld für die Landessprache an, die der Benutzer während der angemeldeten Sitzung verwenden möchte.

Mögliche Werte für die Codes finden Sie in [Codes für Landessprachen](#).

Anmerkung: CICS-Nachrichten werden nur in britischem Englisch, vereinfachtem Chinesisch und Japanisch unterstützt. Falls eine andere als eine dieser drei Sprachen angegeben wird, wird standardmäßig Englisch verwendet.

NATLANGINUSE(datenbereich)

Gibt einen einstelligen Code für die Landessprache an, die während der angemeldeten Sitzung verwendet wird. Die aktuelle Implementierung gibt immer das Zeichen E (für amerikanisches Englisch) zurück, das der in der Option NATLANG angegebenen Sprache entspricht. Die Option NATLANGINUSE entspricht Folgendem (in absteigender Priorität):

- Der Sprache, die in der Option NATLANG des Befehls SIGNON TOKEN angegeben ist.
- Der Sprache, die dem Benutzer zugeordnet ist. Diese Zuordnung ist im LANGUAGE-Segment des externen Sicherheitsmanagers angegeben.
- Die Sprache, die der Definition des Terminals zugeordnet ist.
- Die Sprache, die der Standard-Benutzer-ID für die CICS-Region zugeordnet ist.
- Die Standardsprache, die in den Systeminitialisierungsparametern angegeben ist.

Mögliche Werte für die Codes finden Sie unter [Codes für Landessprachen](#).

TOKEN(datenbereich)

Ein Token, das bei einem Sicherheitstokenservice (STS) angefordert wurde.

TOKENLEN(datenwert)

Gibt die Länge des Tokens in Form eines Vollwort-Binärwertes an.

TOKENTYPE(cvda)

Gibt den Typ des Tokens an.

KERBEROS

Das Token ist ein Kerberos-Token.

Bedingungen

16 INVREQ

RESP2-Werte:

9

Das Terminal ist bereits angemeldet.

10

Dieser Task ist kein Terminal zugeordnet.

11

Für das Terminal dieser Task ist die Sicherheit voreingestellt.

12

Die Antwort von den CICS-Sicherheitsmodulen wurde nicht erkannt.

13

Der vom externen Sicherheitsmanager zurückgegebene Wert in der Option ESMRESP ist nicht von CICS klassifiziert. Eine Erläuterung der Optionen ESMRESP und ESMREASON finden Sie in der ESM-Dokumentation.

14

Die erforderliche Landessprache ist nicht verfügbar.

15

Die Anmeldung wurde unter Verwendung des Transaktionsroutings versucht, ohne dass die Transaktion CRTE verwendet wurde.

18

Die CICS ESM-Schnittstelle wurde nicht initialisiert (als Systeminitialisierungsparameter war SEC=NO angegeben).

25

Das Terminal hat einen ungültigen Typ.

- 26** Während der Überprüfung von SNSCOPE trat ein Fehler auf. Der Grenzwert für MVS ENQ-Anforderungen wurde erreicht.
- 27** Der externe Sicherheitsmanager ist nicht aktiv.
- 28** Die erforderliche Landessprache ist ungültig.
- 29** Der Benutzer ist bereits angemeldet. Dieser Code bezieht sich auf die Überprüfung der Befehlsberechtigung für die Anmeldung.
- 30** Der externe Sicherheitsmanager antwortet nicht.
- 31** Für **TOKENTYPE** wurde ein anderer CVDA-Wert als KERBEROS angegeben.
- 32** Für **DATATYPE** wurde ein anderer CVDA-Wert als BASE64 oder BIT angegeben.
- 36** Der Datentyp BASE64 war angegeben, aber der Wert für **TOKEN** enthält keine Daten in Base64-Codierung.
- 40** Das Key-Distribution-Center wurde nicht gestartet oder wird gerade beendet.
- 41** Das Key-Distribution-Center antwortet nicht.
- 47** Beim externen Sicherheitsmanager ist keine Benutzer-ID für den Kerberos-Prinzipal definiert, der dem Token zugeordnet ist.
- 50** Bei den in **TOKEN** angegebenen Daten handelt es sich nicht um ein Kerberos-Token.
- 200** Der Befehl ist für ein Serverprogramm von Distributed Program Link nicht zulässig.
Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

RESP2-Werte:

- 45** Die Länge des Kerberos-Tokens überschreitet den Maximalwert von 65535.

70 NOTAUTH

RESP2-Werte:

- 16** Die Benutzer-ID ist nicht für die Verwendung des Terminals berechtigt.
- 17** Die Benutzer-ID ist nicht für die Verwendung der Anwendung berechtigt.
- 19** Die Benutzer-ID wurde entzogen.
- 20** Der Zugriff der Benutzer-ID auf die angegebene Gruppe wurde entzogen.
- 21** Die Anmeldung ist während der Überprüfung von SECLABEL fehlgeschlagen.

22

Die Anmeldung ist fehlgeschlagen, weil der externe Sicherheitsmanager gegenwärtig keine Anmeldung akzeptiert.

23

Die Gruppen-ID ist dem externen Sicherheitsmanager nicht bekannt.

24

Die Benutzer-ID gehört nicht zur Gruppen-ID.

40

Der externe Sicherheitsmanager hat die Anforderung zur Verifizierung des Tokens nicht autorisiert. Fehlermeldungen werden an das CSCS-Ziel gesendet.

42

Eine Kerberos-Anforderung kann nicht ausgeführt werden, weil das zugehörige Ticket abgelaufen ist.

43

Der Authentifikator ist abgelaufen.

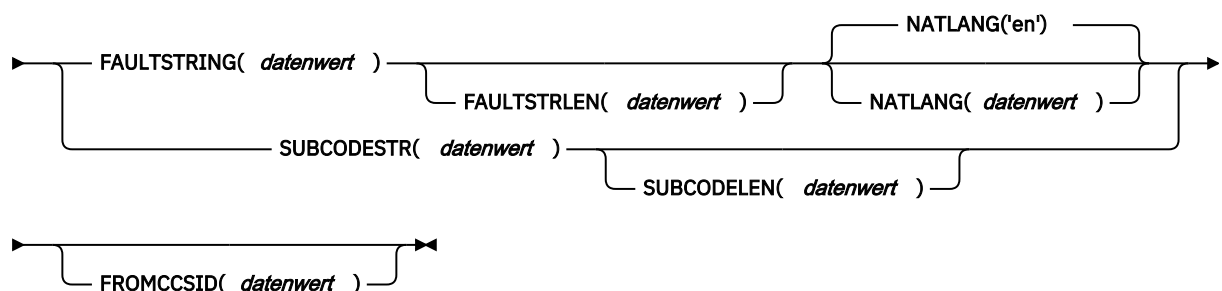
Standardaktion: Die Task wird abnormal beendet.

SOAPFAULT ADD

Fügt Informationen zu einem vorhandenen SOAPFAULT-Objekt hinzu. Sie können diesen Befehl nur in einem Programm verwenden, das über einen von CICS bereitgestellten SOAP-Nachrichtenhandler aufgerufen wird.

SOAPFAULT ADD

➔ SOAPFAULT ADD ➔



Bedingungen: CHANNELERR, INVREQ, LENGERR

Dieser Befehl ist threadsicher.

Beschreibung

Dieser Befehl fügt Informationen zu einem SOAPFAULT-Objekt hinzu, das mit einem früheren Befehl **SOAPFAULT CREATE** erstellt wurde. Die hinzugefügten Informationen können Folgendes umfassen:

- Untercode.
- Fehlerzeichenfolge für eine bestimmte Landessprache. Falls der Fehler bereits eine Fehlerzeichenfolge für die Sprache enthält, wird die Fehlerzeichenfolge für diese Sprache durch diesen Befehl ersetzt. In SOAP 1.1 wird nur die Fehlerzeichenfolge für die Originalsprache verwendet.

Dieser Befehl erfordert Informationen, die in Containern im Kanal des bereitgestellten SOAP-Nachrichtenhandlers vorgehalten werden. Damit Sie diesen Befehl verwenden können, müssen Sie Zugriff auf den Kanal besitzen. Diesen Zugriff besitzen nur die folgenden Typen von Programmen:

- Als SOAP-Header-Handler aufgerufene Programme.
- Direkt aus einem von CICS bereitgestellten SOAP-Nachrichtenhandler aufgerufene Programme.

- Mit dem CICS-Web-Service-Assistenten implementierte Programme, die eine Kanalschnittstelle besitzen. Programme mit einer COMMAREA-Schnittstelle haben *keinen* Zugriff auf den Kanal.
- Programme, die globale Benutzerexits zur Überwachung von Web-Services verwenden.

Viele Optionen dieses Befehls gelten für Fehler in SOAP 1.1 und SOAP 1.2; ihr Verhalten weicht allerdings bei der jeweiligen SOAP-Version etwas ab. Andere Optionen gelten entweder für die eine SOAP-Version oder für die andere SOAP-Version, jedoch nicht für beide Versionen. Falls Sie eine solche Version angeben, wenn die Nachricht eine andere SOAP-Version verwendet, löst der Befehl eine Bedingung INVREQ aus. Damit Sie feststellen können, welche SOAP-Version für die Nachricht gilt, enthält der Container DFHWS-SOAPLEVEL ein binäres Vollwort mit einem der folgenden Werte:

1

Die Anforderung oder Antwort ist eine Nachricht von SOAP 1.1.

2

Die Anforderung oder Antwort ist eine Nachricht von SOAP 1.2.

10

Die Anforderung oder Antwort ist eine keine SOAP-Nachricht.

Optionen

SUBCODESTR (datenwert)

Gibt den Inhalt eines Elements <Subcode> an, das zum SOAPFAULT-Objekt hinzugefügt werden soll. Der Untercode kann bis zu 64 Zeichen lang sein und muss ein qualifizierter XML-Name (QName) sein.

- Bei SOAP 1.1 wird diese Option ignoriert.
- Bei SOAP 1.2 gibt diese Option den Inhalt des Elements <Subcode> an.

SUBCODELEN (datenwert)

Gibt als Vollwort-Binärwert die Länge des Elements <Subcode> an, das in der Option SUBCODESTR angegeben ist.

FAULTSTRING (datenwert)

Gibt eine lesbare Erläuterung des Fehlers an. Der Wert für die Option FAULTSTRING kann bis zu 2056 Zeichen lang sein.

- Bei SOAP 1.1 gibt diese Option den Inhalt des Elements <faultstring> an.
- Bei SOAP 1.2 gibt diese Option den Inhalt des Elements <Reason> an.

FAULTSTRLEN (datenwert)

Gibt als Vollwort-Binärwert die Länge der Option FAULTSTRING an.

FROMCCSID (datenwert)

Gibt als Vollwort-Binärzahl die aktuelle ID des codierten Zeichensatzes (Coded Character Set Identifier, CCSID) für die Zeichendaten an, die in den SOAP-Fehler eingefügt werden sollen. Falls diese Option nicht angegeben ist, verwendet CICS den Wert, der im Systeminitialisierungsparameter **LO-CALCCSID** angegeben ist. Weitere Informationen zu CCSIDs sowie eine Liste der von CICS unterstützten CCSIDs finden Sie unter [Von CICS unterstützte Konvertierungen](#).

NATLANG (datenwert)

Gibt ein 8 Zeichen umfassendes Feld an, das die für den Wert von FAULTSTRING verwendete Landessprache enthält. Die Sprache wird unter Verwendung der Sprachen-ID von XML 1.0 angegeben. Der Standardwert ist 'en' (Englisch).

Wenn die Sprachenkennung kürzer als acht Zeichen ist, müssen Sie sie rechts mit Leerzeichen aus dem Zeichensatz auffüllen, der in der Option FROMCCSID (bzw. dem CICS-Parameter LOCALCCSID) angegeben ist. Falls Sie beispielsweise den UTF-8-Zeichensatz mit FROMCCSID(1208) angeben, müssen Sie den Wert für NATLANG mit Zeichen X'20' auffüllen.

Diese Option wird nur für Fehler von SOAP 1.2 verwendet.

Bedingungen

122 CHANNELERR

Die RESP2-Werte lauten wie folgt:

3

Der Kanal, zu dem dieses SOAPFAULT-Objekt hinzugefügt wird, ist schreibgeschützt.

16 INVREQ

Die RESP2-Werte lauten wie folgt:

3

Der Befehl wurde außerhalb der Umgebung eines von CICS bereitgestellten SOAP-Handlers ausgegeben.

7

Es ist kein SOAP-Fehler vorhanden.

11

Der Untercode ist ungültig.

22 LENGERR

Die RESP2-Werte lauten wie folgt:

6

Der Wert von FAULTSTRLEN ist ungültig.

10

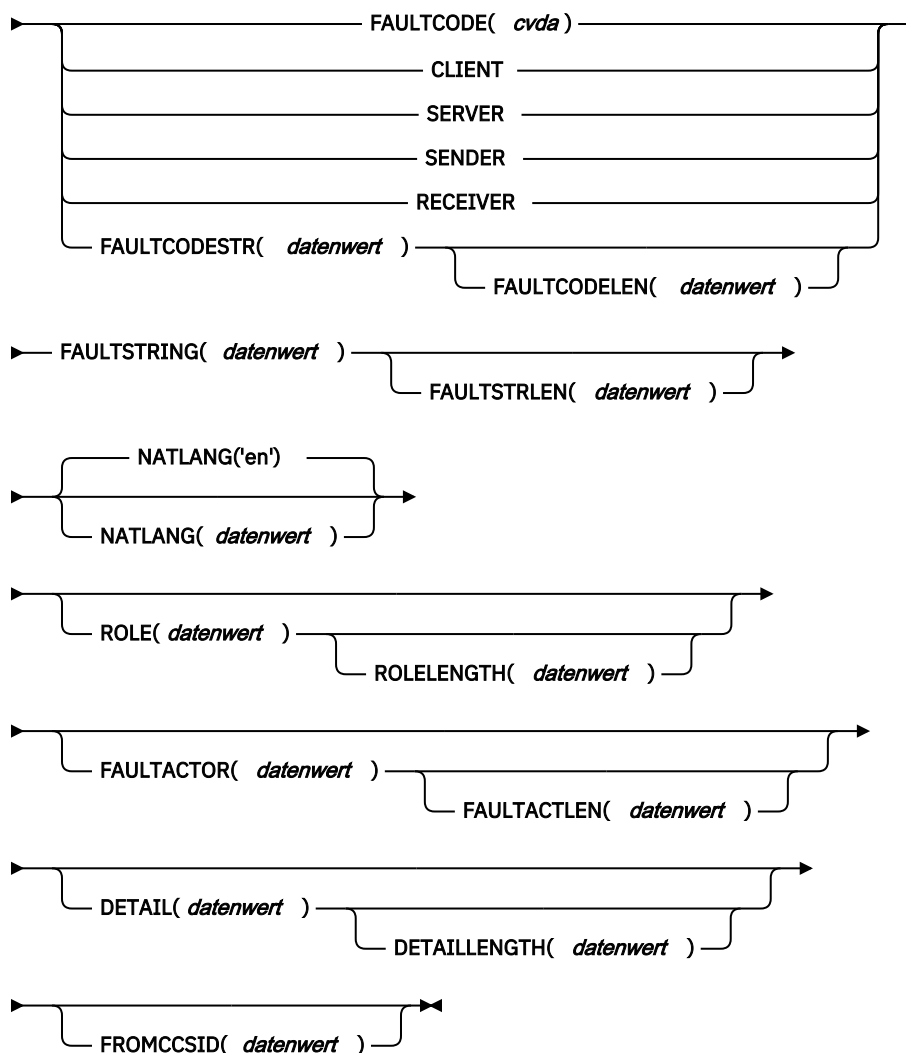
Der Wert von SUBCODELEN ist ungültig.

SOAPFAULT CREATE

Erstellt eine SOAP-Fehlerantwortnachricht, die als Antwort auf eine Web-Service-Anforderung zurückgegeben wird. Die Antwort der Web-Service-Anwendung wird nicht verarbeitet.

SOAPFAULT CREATE

➔ SOAPFAULT CREATE ➔



Bedingungen: CCSIDERR, CHANNELERR, INVREQ, LENGERR

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl **SOAPFAULT CREATE** erstellt einen SOAP-Fehler. Falls es im Kontext der SOAP-Nachricht, die durch den Nachrichtenhandler verarbeitet wird, bereits einen SOAP-Fehler gibt, wird der vorhandene Fehler überschrieben. Dieser Befehl kann nur in Programmen verwendet werden, die von einem bereitgestellten SOAP-Handler angehängt werden. Axis2-Anwendungen, die in einer Java-Pipeline in CICS gehostet werden, können diesen Befehl nicht verwenden.

Dieser Befehl erfordert Informationen, die in Containern im Kanal des bereitgestellten SOAP-Nachrichtenhandlers vorgehalten werden. Damit Sie diesen Befehl verwenden können, müssen Sie Zugriff auf den Kanal besitzen. Diesen Zugriff besitzen nur die folgenden Typen von Programmen:

- Als SOAP-Header-Handler aufgerufene Programme.
- Direkt aus einem von CICS bereitgestellten SOAP-Nachrichtenhandler aufgerufene Programme.
- Mit dem CICS-Web-Service-Assistenten implementierte Programme, die eine Kanalschnittstelle besitzen. Programme mit einer COMMAREA-Schnittstelle haben *keinen* Zugriff auf den Kanal.
- Programme, die globale Benutzerexits zur Überwachung von Web-Services verwenden.

Viele Optionen dieses Befehls gelten für Fehler in SOAP 1.1 und SOAP 1.2; ihr Verhalten weicht allerdings bei der jeweiligen SOAP-Version etwas ab. Andere Optionen gelten entweder für die eine SOAP-Version oder für die andere SOAP-Version, jedoch nicht für beide Versionen. Falls Sie eine solche Version angeben, wenn die Nachricht eine andere SOAP-Version verwendet, löst der Befehl eine Bedingung INVREQ aus. Damit Sie feststellen können, welche SOAP-Version für die Nachricht gilt, enthält der Container DFHWS-SOAPLEVEL ein binäres Vollwort mit einem der folgenden Werte:

1

Die Anforderung oder Antwort ist eine Nachricht von SOAP 1.1.

2

Die Anforderung oder Antwort ist eine Nachricht von SOAP 1.2.

10

Die Anforderung oder Antwort ist eine keine SOAP-Nachricht.

Optionen

DETAIL (datenwert)

Gibt den folgenden Inhalt an:

- Bei SOAP 1.1 gibt diese Option den Inhalt des Elements <detail> des SOAP-Fehlers an.
- Bei SOAP 1.2 gibt diese Option den Inhalt des Elements <Detail> des SOAP-Fehlers an.

Dieses Element sollte eines oder mehrere gültige und durch den Namensbereich qualifizierte XML-Elemente bzw. Leerzeichen enthalten. Eine vollständige Beschreibung des gültigen Inhalts für das Element können Sie den entsprechenden SOAP-Spezifikationen entnehmen.

Das Element überträgt anwendungsspezifische Fehlerinformationen für das Element <Body> und wird verwendet, wenn der Inhalt des Elements <Body> nicht erfolgreich verarbeitet werden konnte. Bei SOAP 1.1 muss das Element <detail> vorhanden sein, wenn der Inhalt des Elements <Body> nicht erfolgreich verarbeitet werden konnte; bei SOAP 1.2 ist das Element <Detail> optional.

Falls der Befehl SOAPFAULT CREATE in einem Header-Handler-Programm ausgegeben wird, wird der Detailinhalt für SOAP 1.1 in einem Headerblock übertragen. Für SOAP 1.2 wird das Element <Detail> im SOAP-Fehler übertragen.

DETAILLENGTH (datenwert)

Gibt als Vollwort-Binärwert die Länge der Option DETAIL an.

FAULTACTLEN (datenwert)

Gibt als Vollwort-Binärwert die Länge der Option FAULTACTOR an.

FAULTACTOR (datenwert)

Gibt den folgenden Inhalt an:

- Bei SOAP 1.1 gibt diese Option den Inhalt des Elements <faultactor> des SOAP-Fehlers an.
- Bei SOAP 1.2 gibt diese Option den Inhalt des Elements <Node> an.

Der Wert für die Option FAULTACTOR kann bis zu 2056 Zeichen lang sein und muss eine gültige URI darstellen (anyURI).

FAULTCODE (cvda)

Die CVDA-Werte lauten wie folgt:

CLIENT

Bei SOAP 1.1 sollte der CVDA-Wert Client angegeben sein. Falls Sie jedoch für eine Fehlerantwortnachricht von SOAP 1.1 Sender angeben, verwendet CICS den CVDA-Wert Client.

SENDER

Bei SOAP 1.2 sollte der CVDA-Wert Sender angegeben sein. Falls Sie jedoch für eine Fehlerantwortnachricht von SOAP 1.2 Client angeben, verwendet CICS den CVDA-Wert Sender.

SERVER

Bei SOAP 1.1 sollte der CVDA-Wert Server angegeben sein. Falls Sie jedoch für eine Fehlerantwortnachricht von SOAP 1.1 Receiver angeben, verwendet CICS den CVDA-Wert Server.

RECEIVER

Bei SOAP 1.2 sollte der CVDA-Wert `Receiver` angegeben sein. Falls Sie jedoch für eine Fehlerantwortnachricht von SOAP 1.2 Server angeben, verwendet CICS den CVDA-Wert `Receiver`.

FAULTCODELEN(*datenwert*)

Gibt als Vollwort-Binärwert die Länge der Option `FAULTCODESTR` an.

FAULTCODESTR(*datenwert*)

Gibt einen benutzerdefinierten SOAP-Fehlercode für eine Nachricht von SOAP 1.1 an. Der Fehlercode kann bis zu 64 Zeichen lang sein und muss ein qualifizierter XML-Name (QName) sein. Die Verwendung des Punktzeichens (.) zum Abgrenzen von Fehlercodewerten wird nicht unterstützt. Bei SOAP 1.1 stellt diese Option den Inhalt des Elements `<faultcode>` bereit.

Falls Sie diese Option für eine Nachricht von SOAP 1.2 festlegen, tritt eine Bedingung `INVREQ` auf.

FAULTSTRING(*datenwert*)

Gibt eine lesbare Erläuterung des Fehlers an. Der Wert für die Option `FAULTSTRING` kann bis zu 2056 Zeichen lang sein.

- Bei SOAP 1.1 gibt diese Option den Inhalt des Elements `<faultstring>` an.
- Bei SOAP 1.2 gibt diese Option den Inhalt des Elements `<Reason>` an.

FAULTSTRLEN(*datenwert*)

Gibt als Vollwort-Binärwert die Länge der Option `FAULTSTRING` an.

FROMCCSID(*datenwert*)

Gibt als Vollwort-Binärzahl die aktuelle ID des codierten Zeichensatzes (Coded Character Set Identifier, CCSID) für die Zeichendaten an, die in den SOAP-Fehler eingefügt werden sollen. Falls diese Option nicht angegeben ist, verwendet CICS den Wert, der im Systeminitialisierungsparameter **LO-CALCCSID** angegeben ist. Weitere Informationen zu CCSIDs sowie eine Liste der von CICS unterstützten CCSIDs finden Sie unter [Von CICS unterstützte Konvertierungen](#).

NATLANG(*datenwert*)

Gibt ein 8 Zeichen umfassendes Feld an, das die für den Wert von `FAULTSTRING` verwendete Landessprache enthält. Die Sprache wird unter Verwendung der Sprachen-ID von XML 1.0 angegeben. Der Standardwert ist 'en' (Englisch).

Wenn die Sprachenkennung kürzer als acht Zeichen ist, müssen Sie sie rechts mit Leerzeichen aus dem Zeichensatz auffüllen, der in der Option `FROMCCSID` (bzw. dem CICS-Parameter `LOCALCCSID`) angegeben ist. Falls Sie beispielsweise den UTF-8-Zeichensatz mit `FROMCCSID(1208)` angeben, müssen Sie den Wert für `NATLANG` mit Zeichen `X'20'` auffüllen.

Diese Option wird nur für Fehler von SOAP 1.2 verwendet.

ROLE(*datenwert*)

Gibt die URI an, die die Rolle des SOAP-Knotens beschreibt, von dem der Fehler generiert wurde. Der Wert für die Option `ROLE` kann bis zu 2056 Zeichen lang sein und muss eine gültige URI darstellen (XML-Typ `anyURI`).

- Bei SOAP 1.1 wird diese Option ignoriert.
- Bei SOAP 1.2 gibt diese Option den Inhalt des Elements `<Role>` an.

ROLELENGTH(*datenwert*)

Gibt als Vollwort-Binärwert die Länge der Option `ROLE` an.

Bedingungen

123 CCSIDERR

Die `RESP2`-Werte lauten wie folgt:

13

Es wurde eine ungültige CCSID angegeben.

14

Es wurde eine nicht unterstützte CCSID angegeben.

122 CHANNELERR

Die RESP2-Werte lauten wie folgt:

3

Der Kanal, in dem dieses SOAPFAULT-Objekt erstellt wird, ist schreibgeschützt.

16 INVREQ

Die RESP2-Werte lauten wie folgt:

3

Der Befehl wurde außerhalb der Umgebung eines von CICS bereitgestellten SOAP-Handlers ausgegeben.

11

Der angegebene Wert von FAULTCODE ist ungültig oder für einen Fehler von SOAP 1.2 wurde die Option FAULTCODESTR angegeben.

13

Die Option DETAIL enthält weder ein gültiges mit Namensbereich qualifiziertes XML-Element noch Leerzeichen.

22 LENGERR

Die RESP2-Werte lauten wie folgt:

5

Der Wert der Option FAULTCODELEN ist ungültig.

6

Der Wert der Option FAULTSTRLEN ist ungültig.

7

Der Wert der Option ROLELENGTH ist ungültig.

8

Der Wert der Option FAULTACTLEN ist ungültig.

9

Der Wert der Option DETAILLENGTH ist ungültig.

SOAPFAULT DELETE

Löscht ein vorhandenes SOAPFAULT-Objekt. Sie können diesen Befehl nur in einem Programm verwenden, das über einen von CICS bereitgestellten SOAP-Nachrichtenhandler aufgerufen wird.

SOAPFAULT DELETE

➡ SOAPFAULT DELETE ➡

Bedingungen: CHANNELERR, INVREQ, NOTFND

Dieser Befehl ist threadsicher.

Beschreibung

Dieser Befehl löscht ein SOAPFAULT-Objekt, das mit einem früheren Befehl **SOAPFAULT CREATE** erstellt wurde.

Dieser Befehl erfordert Informationen, die in Containern im Kanal des bereitgestellten SOAP-Nachrichtenhandlers vorgehalten werden. Damit Sie diesen Befehl verwenden können, müssen Sie Zugriff auf den Kanal besitzen. Diesen Zugriff besitzen nur die folgenden Typen von Programmen:

- Als SOAP-Header-Handler aufgerufene Programme.
- Direkt aus einem von CICS bereitgestellten SOAP-Nachrichtenhandler aufgerufene Programme.
- Mit dem CICS-Web-Service-Assistenten implementierte Programme, die eine Kanalschnittstelle besitzen. Programme mit einer COMMAREA-Schnittstelle haben *keinen* Zugriff auf den Kanal.
- Programme, die globale Benutzerexits zur Überwachung von Web-Services verwenden.

Bedingungen

122 CHANNELERR

3

Der Kanal, in dem dieses SOAPFAULT-Objekt gelöscht wird, ist schreibgeschützt.

16 INVREQ

Die RESP2-Werte lauten wie folgt:

3

Die Funktion wurde aufgerufen, als ein von CICS bereitgestellter SOAP-Knoten nicht verwendet wurde.

13 NOTFND

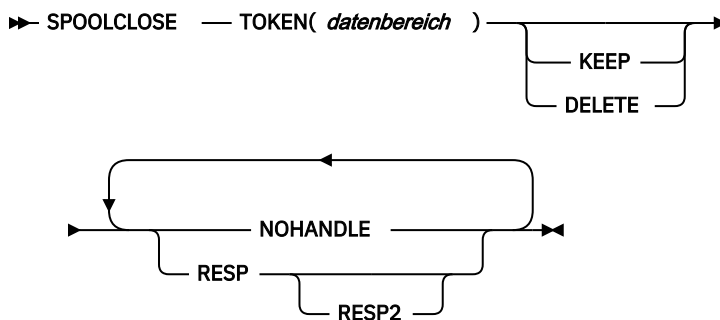
2

Es ist kein SOAPFAULT-Objekt vorhanden, das gelöscht werden kann.

SPOOLCLOSE

Schließt einen Spoolbericht.

SPOOLCLOSE TOKEN



Bedingungen: ALLOCERR, INVREQ, NOSPOOL, NOSTG, NOTFND, NOTOPEN, STRELERR

Beschreibung

Der Befehl SPOOLCLOSE schließt einen CICS-Spoolbericht und ändert optional dessen Aufbewahrungsmerkmale. Falls mehrere Transaktionen gleichzeitig versuchen, Berichte aus JES zu lesen, sollte auf den Befehl **nicht** direkt ein Befehl SPOOLOPEN folgen. Auf diesen Befehl sollte ein Befehl WAIT folgen, damit andere Transaktionen die Schnittstelle verwenden können.

Falls im Befehl SPOOLCLOSE weder KEEP noch DELETE angegeben ist oder falls der Bericht implizit durch einen Befehl SYNCPOINT oder RETURN geschlossen wird, wird eine Standarddisposition angenommen:

- Wenn ein Eingabebericht explizit durch einen Befehl SPOOLCLOSE geschlossen wird, lautet die Standarddisposition DELETE.
- In allen Fällen lautet die Standarddisposition KEEP.

Optionen

DELETE

Für einen Eingabebericht gibt die Option DELETE an, dass beim nachfolgenden Befehl OPEN INPUT der **nächste** Bericht gelesen werden soll.

Für einen Ausgabebericht gibt die Option DELETE an, dass der Bericht gelöscht werden soll.

Anmerkung: Wenn unter Verwendung des internen Lesers (INTRDR) unter Angabe der Option DELETE ein JCL-Job übergeben wird, wird der Job bisweilen vor dem Löschen der Ausgabe ausgeführt.

KEEP

Für einen Eingabebericht gibt die Option KEEP an, dass der Bericht bei der nächsten Ausgabe des Befehls SPOOLOPEN INPUT erneut gelesen werden soll.

Für einen Ausgabebericht gibt die Option KEEP an, dass der Bericht an seinen Zielknoten gesendet werden soll.

TOKEN(*datenbereich*)

Gibt ein 8 Zeichen umfassendes und von CICS zugeordnetes Token für die Kennzeichnung eines Berichtes an.

Bedingungen

Anmerkung: Es gibt keine Standardaktionen.

85 ALLOCERR

Tritt in einer der folgenden Situationen auf:

- Die dynamische Zuordnung hat eine Anforderung zum Zuordnen einer Eingabedatei zurückgewiesen.

Das Feld RESP2 gibt den Antwortcode für die dynamische Zuordnung an, der diesen Fehler bezeichnet. Die ersten beiden Zeichen sind der Informationsursachencode (S99INFO) und die zweiten zwei Zeichen sind der Fehlerursachencode (S99ERROR); eine Definition finden Sie in der Veröffentlichung [z/OS MVS Programming: Authorized Assembler Services Guide](#).

16 INVREQ

RESP2-Werte:

4

Nicht unterstützte Sprache.

8

Nicht unterstützte Funktion.

40

Die Subsystemschnittstelle ist bereits aktiviert.

Anmerkung: Die Fehler 1024 und höher sind interne Fehler, die nicht auftreten sollten. Falls einer dieser Fehlercodes zurückgegeben wird, wenden Sie sich an Ihr zuständiges IBM Support Center.

80 NOSPOOL

RESP2-Werte:

4

Es ist kein Subsystem vorhanden.

8

Die Schnittstelle wird inaktiviert; CICS wird in den Wartemodus versetzt.

12

Die Schnittstelle wurde gestoppt.

42 NOSTG

Tritt in einer der folgenden Situationen auf:

- Ein Befehl GETMAIN ist innerhalb der JES-Schnittstellensubtask (DFHPSPSS) fehlgeschlagen.

Das Feld RESP2 gibt den Rückgabecode 15 für das GETMAIN-Register zurück.

13 NOTFND

RESP2-Werte:

1024

Die Eingabe- oder Ausgabefunktion wurde beschädigt und der Befehl SPOOLCLOSE konnte nicht vollständig ausgeführt werden.

19 NOTOPEN

RESP2-Werte:

Die Datei wurde nicht geöffnet.

86 STRELERR

Tritt in einer der folgenden Situationen auf:

- Ein Befehl FREEMAIN ist innerhalb der JES-Schnittstellensubtask (DFHPSPSS) fehlgeschlagen.

Das Feld RESP2 gibt den Rückgabecode 15 für das FREEMAIN-Register zurück.

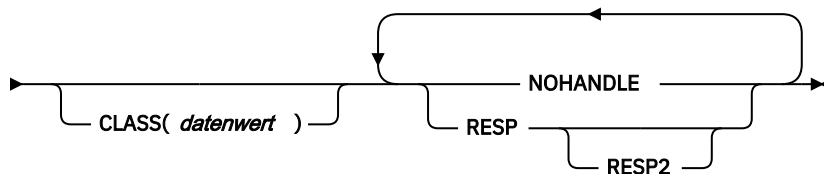
Einschränkung: Die Option RESP oder NOHANDLE muss im EXEC CICS-Befehl **SPOOLCLOSE** angegeben werden.

SPOOLOPEN INPUT

Öffnet einen Spoolbericht.

SPOOLOPEN INPUT

➔ SPOOLOPEN INPUT — TOKEN(*datenbereich*) — USERID(*datenwert*) ➔



Bedingungen: ALLOCERR, ILOGIC, INVREQ, NOSPOOL, NOSTG, NOTAUTH, NOTFND, NOTOPEN, OPENERR, SPOLBUSY, SPOLERR, STRELERR

Beschreibung

Der Befehl SPOOLOPEN INPUT öffnet einen Spoolbericht für die Eingabe aus dem Systemspooler in CICS.

Er bereitet den direkten Abruf (Lesevorgang) einer vorhandenen Spooldatei mithilfe eines externen Ausgabeprogrammnamens (USERID) und einer angegebenen Klasse vor.

Eine andere Task könnte eine Spooldatei für die Eingabe zugeordnet haben. In diesem Fall sollten Sie den Befehl nach einem angemessenen Zeitintervall wiederholen.

Nachdem dieser Befehl erfolgreich ausgeführt wurde, sollten Sie den Bericht so bald wie möglich lesen und mit CLOSE fortfahren, damit andere Benutzer den JES-Einzelthread nutzen können. Falls der Befehl SPOOLCLOSE nicht vor dem Transaktionsende oder einem Synchronisationspunkt ausgegeben wird, führt CICS einen impliziten Befehl SPOOLCLOSE KEEP aus und schreibt eine Nachricht an CSMT, um den Systemprogrammierer über die mögliche unnötige Beibehaltung von Ressourcen zu informieren. Sie sollten den Befehl SPOOLOPEN für eine Datei nur dann ausführen, wenn Sie darauf vorbereitet sind, ihn vollständig zu verarbeiten.

Wenn dieser Befehl erfolgreich ausgeführt wird, gibt er ein Token zurück, mit dem der Bericht später in Befehlen SPOOLREAD und SPOOLCLOSE angegeben wird.

Optionen

CLASS(*datenwert*)

Gibt eine einstellige Klassenbezeichnung an. Der Operand CLASS kann als Auswahlparameter für Eingabeberichte verwendet werden. Falls er nicht angegeben ist, wird ungeachtet seiner Klasse der erste Bericht für die mit USERID angegebene Benutzer-ID abgerufen.

TOKEN(*datenbereich*)

Gibt ein 8 Zeichen umfassendes und von CICS zugeordnetes Token für die Kennzeichnung eines Berichtes an.

USERID(datenwert)

Gibt die 8 Zeichen lange Benutzer-ID an. Sie muss mit denselben 4 Zeichen wie die generische CICS-Anwendungs-ID beginnen, damit CICS überprüfen kann, dass Benutzer nicht versuchen, auf Dateien zuzugreifen, die nicht für ihr CICS-System bestimmt sind.

Bedingungen

Anmerkung: Es gibt keine Standardaktionen.

85 ALLOCERR

Tritt in einer der folgenden Situationen auf:

- Die dynamische Zuordnung hat eine Anforderung zum Zuordnen einer Eingabedatei zurückgewiesen.

Das Feld RESP2 gibt den Antwortcode für die dynamische Zuordnung an, der diesen Fehler bezeichnet. Die ersten beiden Zeichen sind der Informationsursachencode (S99INFO) und die zweiten zwei Zeichen sind der Fehlerursachencode (S99ERROR); eine Definition finden Sie in der Veröffentlichung [z/OS MVS Programming: Authorized Assembler Services Guide](#).

21 ILLOGIC

RESP2-Werte:

3

Es wurde ein ungültiger Wert für CLASS angegeben.

16 INVREQ

RESP2-Werte:

4

Nicht unterstützte Sprache.

8

Nicht unterstützte Funktion.

16

Der Wert für USERID fehlt.

36

Der Wert für INPUT|OUTPUT fehlt.

40

Die Subsystemschnittstelle ist bereits aktiviert.

Anmerkung: Die Fehler 1024 und höher sind interne Fehler, die nicht auftreten sollten. Falls einer dieser Fehlercodes zurückgegeben wird, wenden Sie sich an Ihr zuständiges IBM Support Center.

80 NOSPOOL

RESP2-Werte:

4

Es ist kein Subsystem vorhanden.

8

Die Schnittstelle wird inaktiviert; CICS wird in den Wartemodus versetzt.

12

Die Schnittstelle wurde gestoppt.

42 NOSTG

Tritt in einer der folgenden Situationen auf:

- Ein Befehl GETMAIN ist innerhalb der JES-Schnittstellensubtask (DFHPSPSS) fehlgeschlagen.

Das Feld RESP2 gibt den Rückgabecode 15 für das GETMAIN-Register zurück.

70 NOTAUTH

Tritt in einer der folgenden Situationen auf:

- Eine Anwendung hat einen Befehl SPOOLOPEN INPUT mit einer nicht autorisierten Benutzer-ID ausgegeben. Damit die Benutzer-ID autorisiert ist, müssen ihre ersten vier Zeichen mit den ersten vier Zeichen der aktuellen CICS-Anwendungs-ID übereinstimmen.

13 NOTFND

RESP2-Werte:

4

Es wurden keine Dateien für den Abruf durch den angegebenen externen Ausgabeprogrammen gefunden oder die Datei ist zwar vorhanden, befindet sich jedoch im Status HELD.

Diese Bedingung kann außerdem zurückgegeben werden, falls die Benutzer-ID für die CICS-Region keinen Zugriff des Typs ALTER auf das entsprechende Profil in der Klasse JESSPOOL besitzt.

1024

Die Eingabe- oder Ausgabefunktion wurde beschädigt und der Befehl SPOOLCLOSE konnte nicht vollständig ausgeführt werden.

19 NOTOPEN

RESP2-Werte:

8

Die Datei wurde nicht geöffnet oder eine Task, die nicht den Befehl SPOOLOPEN für eine Spooldatei ausgegeben hat, hat versucht, auf die Datei zuzugreifen.

1024

Beim Öffnen des Makros OPEN für die Subtask trat ein Fehler auf.

87 OPENERR

RESP2-Werte:

4

Ein VSAM-Makro SHOWCB konnte die Längenwerte der VSAM-Steuerblöcke nicht zurückgeben, die für den Zugriff auf die JES-Spooldatei verwendet werden.

Tritt außerdem in einer der folgenden Situationen auf (RESP2 nicht festgelegt):

- Während der Verarbeitung von SPOOLOPEN trat ein interner Fehler auf, der den Fehlschlag der Anforderung erzwungen hat.

88 SPOLBUSY

RESP2-Werte:

4

Die Schnittstelle wird bereits von einer anderen Task verwendet.

8

Die Schnittstelle wird bereits von der aktuellen Task verwendet.

Tritt außerdem in einer der folgenden Situationen auf (RESP2 nicht festgelegt):

- Der Einzelthread für JES/Eingabe in der JES-Schnittstelle war nicht verfügbar.

89 SPOLERR

Tritt in einer der folgenden Situationen auf:

- Das Makro für die Subsystemschnittstelle (IEFSSREQ) ist fehlgeschlagen. Es wurde kein Eingabedateiname ausgewählt.

Das Feld RESP2 gibt den Antwortcode IEFSSREQ an.

86 STRELERR

Tritt in einer der folgenden Situationen auf:

- Ein Befehl FREEMAIN ist innerhalb der JES-Schnittstellensubtask (DFHPSPSS) fehlgeschlagen.

Das Feld RESP2 gibt den Rückgabecode 15 für das FREEMAIN-Register zurück.

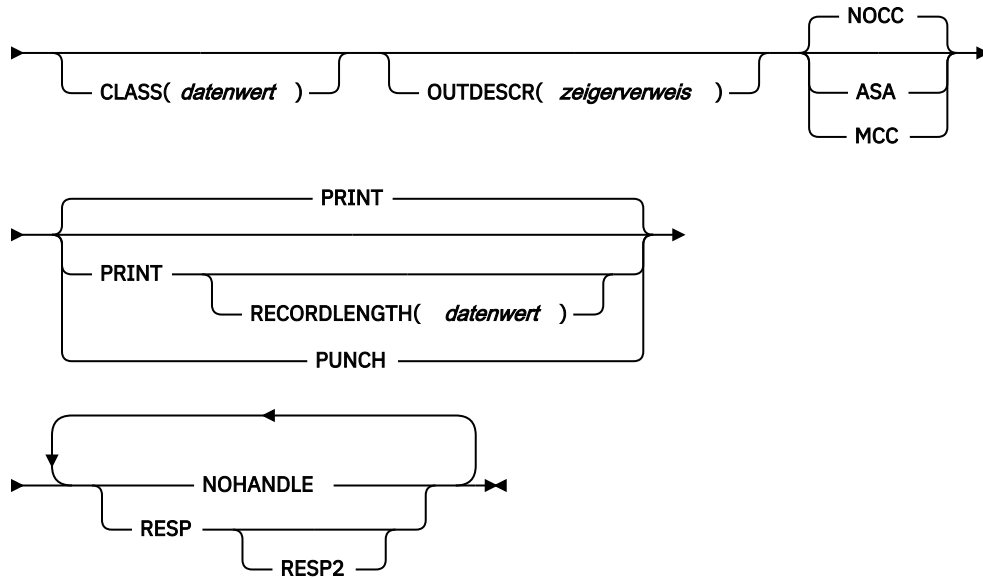
Einschränkung: Die Option RESP oder NOHANDLE muss im EXEC CICS-Befehl **SPOOLOPEN** angegeben werden.

SPOOLOPEN OUTPUT

Öffnet einen Spoolbericht für die Ausgabe.

SPOOLOPEN OUTPUT

➔ SPOOLOPEN OUTPUT — TOKEN(*datenbereich*) — USERID(*datenwert*) — NODE(*datenwert*) —➔



Bedingungen: ALLOCERR, ILOGIC, INVREQ, LENGERR, NODEIDERR, NOSPOOL, NOSTG, NOTFND, NOT-OPEN, OPENERR, OUTDESCERR, SPOLBUSY, STRELERR

Beschreibung

Der Befehl SPOOLOPEN OUTPUT öffnet einen Spoolbericht für die Ausgabe von CICS an den System-Spooler und definiert dessen Merkmale.

Er bewirkt die dynamische Zuordnung der Ausgabedatei unter Verwendung der Knoten-ID, um das ferne Ziel anzugeben, sowie unter Verwendung der Benutzer-ID, um den fernen Benutzer anzugeben. Da es sich hierbei um eine Multithread-Ausgabeanforderung handelt, könnte es zu verzahnten Anforderungen für diesen Service kommen. Dieser Befehl SPOOLOPEN OUTPUT ermöglicht Benutzern die Anforderung eines Tokens für einen Bericht, der voraussichtlich erstellt (geschrieben) wird. Mithilfe dieses Tokens wird der Bericht später in Befehlen SPOOLWRITE und SPOOLCLOSE angegeben.

Verwenden Sie beim Drucken auf einer lokalen Einheit die Optionen NOCC|ASA|MCC, um die Ausgabeformatierung zu steuern. Falls Sie kein Format angeben, wird der Standardwert NOCC verwendet. Mit NODE und USERID kann die Datei nur dann direkt in die lokale Spooldatei geschrieben werden, wenn hierfür der Wert '*' angegeben wird.

Falls Sie den Befehl SPOOLCLOSE nicht vor dem Ende der Transaktion ausgeben, führt CICS einen impliziten Befehl SPOOLCLOSE DELETE aus und schreibt eine Nachricht an CSMT, um Sie über die mögliche unnötige Beibehaltung von Ressourcen zu informieren.

Anmerkung: Falls Sie eine formatierte Datei abrufen, könnte der System-Spooler das Dateiformat geändert haben. Der System-Spooler könnte beispielsweise während der Dateierstellung eine Datei im MCC-Format in das ASA-Format konvertiert haben. Auf die endgültige Druckausgabe hat dies keinen Einfluss.

Optionen

ASA

Gibt an, dass im Bericht jeder Datensatz als Präfix ein ASA-Vorschubsteuerungszeichen besitzt und dass dieses Zeichen vom Betriebssystem verwendet werden muss, um die Formatierung beim Drucken des Berichts zu steuern.

CLASS(datenwert)

Gibt eine einstellige Klassenbezeichnung an. Wenn diese Option weggelassen wird, wird die Klasse A angenommen.

MCC

Gibt an, dass im Bericht jeder Datensatz als Präfix ein Vorschubsteuerungszeichen aus dem IBM Machine Command Code (MCC) besitzt und dass dieses Zeichen vom Betriebssystem verwendet werden muss, um die Formatierung beim Drucken des Berichts zu steuern.

NOCC

Gibt an, dass der Bericht keine internen Formatierungssteuerungen enthält. Wenn der Bericht gedruckt wird, verwendet das Betriebssystem für jeden Datensatz ein Vorschubsteuerungszeichen als Präfix, das den Seitenvorschub gemäß dem Standardwert des Betriebssystems für die Anzahl der Zeilen pro Seite bewirkt.

NODE(datenwert)

Gibt die achtstellige ID eines Zielknotens an, den der System-Spooler für die Weiterleitung der Datei verwendet. Dies ist ein Senderfeld. Falls Sie die lokale Spooldatei angeben wollen und den Operanden OUTDESCR aktivieren, um die Operanden NODE und USERID zu überschreiben, codieren Sie NODE(*) und ebenfalls USERID(*). (Verwenden Sie NODE(*) nicht mit einer anderen Benutzer-ID.) Codieren Sie andernfalls den tatsächlichen Knoten, bei dem es sich um den Namen des Betriebssystems handelt (z. B. MVS, VM), unter dem dieses System für VTAM in dem MVS-System bekannt ist, in dem CICS ausgeführt wird. NODE(LOCAL) ist ebenfalls eine gültige Spezifikation.

Für den Wert von NODE wird eine Gültigkeitsprüfung ausgeführt. Es wird überprüft, ob Leerzeichen (X'40') und Nullen (X'00') vorhanden sind.

OUTDESCR(zeigerverweis)

(Nur bei MVS/SP-JES2 Version 3 oder JES3 Version 4.2.1 bzw. einem höheren aufwärtskompatiblen Release) Gibt eine Zeigervariable an, die auf die Adresse eines Feldes gesetzt werden soll, das die Adresse einer Zeichenfolge von Parametern für die Anweisung OUTPUT von JCL enthält. Dies wird als 'doppelte indirekte Adressierung' bezeichnet. Der Benutzer muss den Zeiger, das Adressfeld und die Zeichenfolge einrichten. Dies bedeutet, dass die Option OUTDESCR nicht innerhalb des Befehlsinterpreters (CECI) verwendet werden kann. Die Zeichenfolge hat das folgende Format:

```
Relative Position - Länge -
Inhalt
0 - 4 - Länge (n) der folgenden Textzeichenfolge
4 - n - Parameter für die Anweisung OUTPUT
```

Die Parameter verwenden dieselben Schlüsselwörter und Werte wie die Anweisung OUTPUT, lediglich die Syntax variiert etwas. Das folgende Format gilt für die Parameterzeichenfolge von OUTDESCR:

```
schlüsselwort1(wert1) [schlüsselwort2(wert2)]
[schlüsselwort3(wert3,wert4)] ...
```

Dies entspricht der folgenden Parameterzeichenfolge für die Anweisung OUTPUT:

```
schlüsselwort1=wert1 [schlüsselwort2=wert2]
[schlüsselwort3=(wert3,wert4)] ...
```

Details über gültige Schlüsselwörter und Werte enthält die Veröffentlichung [z/OS TSO/E System Programming Command Reference](#).

Für den Operanden OUTDESCR gilt Folgendes:

- Er kann die Operanden NODE und USERID nur dann überschreiben, wenn für diese Operanden der Wert '*' angegeben ist.
- Er kann den Operanden CLASS selbst dann nicht überschreiben, wenn dieser nicht angegeben ist und standardmäßig die Klasse A angenommen wird.

Verwenden Sie diesen Operanden, um zusätzliche Attribute für die Spooldatei festzulegen.

PRINT

Ermöglicht das Schreiben von umfangreichen Datensätzen (maximal 32760 Byte) im Spool. Dies ist die Standardeinstellung. Sie ist aus Gründen der Kompatibilität mit der von CICS Transaction Server for z/OS bereitgestellten Spoolunterstützung enthalten.

PUNCH

Muss angegeben werden, falls der Parameter CLASS für die Ausgabedatei einen Stanzer impliziert und die Datei für einen VM/RSCS-Knoten bestimmt ist. Dies stellt sicher, dass der Satzlängenanzeiger auf 80 gesetzt wird, was eine Voraussetzung von VM/RSCS für Stanzdateien ist.

RECORDLENGTH(datenwert)

Gibt als Halbwort-Binärzahlvariable die maximale Länge eines Datensatzes an, der in eine Druckdatei geschrieben werden soll. Der Standardwert ist 32760.

TOKEN(datenbereich)

Gibt ein 8 Zeichen umfassendes und von CICS zugeordnetes Token für die Kennzeichnung eines Berichtes an.

USERID(datenwert)

Gibt die achtstellige ID der Ziel-Benutzer-ID an, die den Bericht verarbeitet. Der Bericht überträgt diese ID, die zur Auswahl des Berichts an seinem Ziel verwendet wird. Dies ist ein Senderfeld, das mit einer Länge von 8 Zeichen deklariert sein muss.

Falls Sie die lokale Spooldatei angeben wollen und den Operanden OUTDESCR aktivieren, um die Operanden NODE und USERID zu überschreiben, codieren Sie USERID('*') und ebenfalls NODE('*'). Codieren Sie andernfalls die tatsächliche Benutzer-ID. Die Bedeutung der Benutzer-ID variiert abhängig vom Betriebssystem. Bei VM ist dies ein bestimmter Benutzer; bei MVS kann es sich um ein JES-externes Ausgabeprogramm oder um ein anderes JES-Ziel, einen TSO-Benutzer bzw. einen anderen auf diesem System ausgeführten Job handeln. Eines solcher Ziele ist der interne JES-Leser, der normalerweise den reservierten Namen INTRDR besitzt. Falls Sie eine tatsächliche Benutzer-ID codieren, verwenden Sie nicht die Angabe NODE('*'), sondern codieren Sie stattdessen den tatsächlichen Knoten.

Der Parameter USERID ist äquivalent zum Parameter WRITER in JES.

Für den Wert von USERID wird eine Gültigkeitsprüfung ausgeführt. Es wird überprüft, ob Leerzeichen (X'40') und Nullen (X'00') vorhanden sind.

Puffer des internen Lesers direkt an JES senden: Statt darauf zu warten, dass der Puffer in Ihrem Adressraum gefüllt wird, senden Sie den Inhalt des Puffers für den internen Leser direkt an JES, indem Sie Ihren letzten Datensatz wie folgt codieren:

```
/*EOF
```

Diese Steueranweisung begrenzt den Job in der Datei und macht ihn für die sofortige Verarbeitung auswählbar.

Weitere Informationen zur Verwendung des internen Lesers und zu anderen Steueranweisungen /* enthält die Veröffentlichung [z/OS MVS JCL User's Guide](#).

Einschränkung: Die Option RESP oder NOHANDLE muss im EXEC CICS-Befehl **SPOOLOPEN** angegeben werden.

Bedingungen

Anmerkung: Es gibt keine Standardaktionen.

85 ALLOCERR

Tritt in einer der folgenden Situationen auf:

- Die dynamische Zuordnung hat eine Anforderung zum Zuordnen einer Eingabedatei zurückgewiesen.

Das Feld RESP2 gibt den Antwortcode für die dynamische Zuordnung an, der diesen Fehler bezeichnet. Die ersten beiden Zeichen sind der Informationsursachencode (S99INFO) und die zweiten zwei

Zeichen sind der Fehlerursachencode (S99ERROR); eine Definition finden Sie in der Veröffentlichung [z/OS MVS Programming: Assembler Services Guide](#).

21 ILLOGIC

Tritt in einer der folgenden Situationen auf:

- Es wurde ein ungültiger Wert für CLASS angegeben.

16 INVREQ

RESP2-Werte:

4

Nicht unterstützte Sprache.

8

Nicht unterstützte Funktion.

16

Der Wert für USERID fehlt.

20

Der Wert für NODE fehlt.

36

Der Wert für INPUT|OUTPUT fehlt.

40

Die Subsystemschnittstelle ist bereits aktiviert.

Anmerkung: Die Fehler 1024 und höher sind interne Fehler, die nicht auftreten sollten. Falls einer dieser Fehlercodes zurückgegeben wird, wenden Sie sich an Ihr zuständiges IBM Support Center.

44

Die Zeichenfolge für OUTDESCR enthält einen Fehler.

48

OUTDESCR ist angegeben, aber die Funktion ist nicht verfügbar (falsche Version von CICS oder JES).

52

OUTDESCR war angegeben, aber es wurde ein falscher Zeiger im Schlüsselwort oder in der Bedingung für OUTDESCR gefunden.

22 LENGERR

Tritt in einer der folgenden Situationen auf:

- Der Wert von RECORDLENGTH liegt nicht zwischen 0 und 32760. Im Feld RESP2 ist der falsche Wert angegeben.

90 NODEIDERR

Tritt in einer der folgenden Situationen auf:

- JES erkennt die Kombination aus NODE/USERID nicht, die im Befehl SPOOL OPEN OUTPUT angegeben ist.

Das Feld RESP2 gibt den Antwortcode für die dynamische Zuordnung an, der diesen Fehler bezeichnet. Die ersten beiden Zeichen sind der Informationsursachencode (S99INFO) und die zweiten zwei Zeichen sind der Fehlerursachencode (S99ERROR); eine Definition finden Sie in der Veröffentlichung [z/OS MVS Programming: Assembler Services Guide](#).

80 NOSPOOL

RESP2-Werte:

4

Es ist kein Subsystem vorhanden.

8

Die Schnittstelle wird inaktiviert; CICS wird in den Wartemodus versetzt.

12

Die Schnittstelle wurde gestoppt.

42 NOSTG

Tritt in einer der folgenden Situationen auf:

- Ein Befehl GETMAIN ist innerhalb der JES-Schnittstellensubtask (DFHPSPSS) fehlgeschlagen.
Das Feld RESP2 gibt den Rückgabecode 15 für das GETMAIN-Register zurück.
- Es war nicht genügend Speicher verfügbar, um die Argumentzeichenfolge für OUTDESCR zu kopieren und zu ändern.

13 NOTFND

RESP2-Werte:

4

Es wurden keine Dateien für den Abruf durch den angegebenen externen Ausgabeprogrammen gefunden.

19 NOTOPEN

RESP2-Werte:

8

Die Datei wurde nicht geöffnet.

1024

Beim Öffnen des Makros OPEN für die Subtask trat ein Fehler auf.

87 OPENERR

RESP2-Werte:

4

Ein VSAM-Makro SHOWCB konnte die Längenwerte der VSAM-Steuerblöcke nicht zurückgeben, die für den Zugriff auf die JES-Spooldatei verwendet werden.

Tritt außerdem in einer der folgenden Situationen auf (RESP2 nicht festgelegt):

- Während der Verarbeitung von SPOOLOPEN trat ein interner Fehler auf, der den Fehlschlag der Anforderung erzwungen hat.

96 OUTDESCRERR

Tritt in einer der folgenden Situationen auf:

- Das Makro OUTADD oder OUTDEL (aufgerufen infolge der Angabe von OUTDESCR) ist fehlgeschlagen.

Das Feld RESP2 enthält den Ursachencode aus dem Makro OUTADD oder OUTDEL. Eine Beschreibung der Codes enthält die Veröffentlichung [z/OS MVS Programming: Assembler Services Guide](#).

88 SPOLBUSY

RESP2-Werte:

4

Die Schnittstelle wird bereits von einer anderen Task verwendet.

8

Die Schnittstelle wird bereits von der aktuellen Task verwendet.

Tritt auch in der folgenden Situation auf:

- Der Einzelthread für JES/Eingabe in der JES-Schnittstelle war nicht verfügbar.

86 STRELERR

Tritt in der folgenden Situation auf:

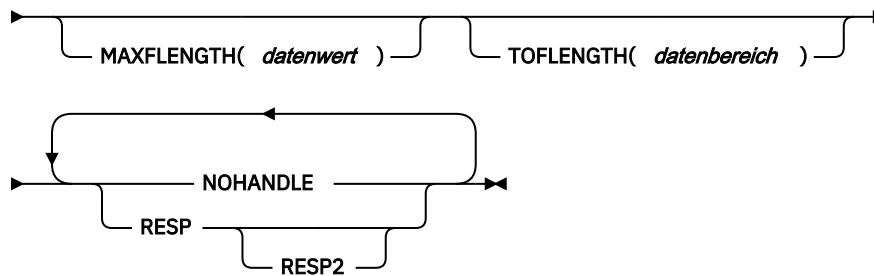
- Ein Befehl FREEMAIN ist innerhalb der JES-Schnittstellensubtask (DFHPSPSS) fehlgeschlagen.
Das Feld RESP2 gibt den Rückgabecode 15 für das FREEMAIN-Register zurück.

SPOOLREAD

Der Befehl SPOOLREAD ruft den nächsten Datensatz vom System-Spooler ab.

SPOOLREAD

➔ SPOOLREAD — TOKEN(*datenbereich*) — INTO(*datenbereich*) ➔



Bedingungen: ALLOCERR, ENDFILE, ILOGIC, INVREQ, LENGERR, NOSPOOL, NOSTG, NOTFND, NOT-OPEN, SPOLBUSY, SPOLERR, STRELERR

Optionen

INTO(*datenbereich*)

Gibt den Datenbereich für die Daten mit variabler Länge an. Dies ist ein Empfängerfeld.

MAXLENGTH(*datenwert*)

Gibt als Vollwort-Binärzahlvariable die maximale Länge der übertragenen Daten an. Dieser Wert wird durch den Benutzer bei der Eingabe festgelegt. Der Grenzwert für die **Länge** beträgt 32760 Byte. Dies ist die maximale Größe des CICS-Puffers, der zum Lesen eines Datensatzes verwendet wird.

TOLENGTH(*datenbereich*)

Gibt als Vollwort-Binärzahlvariable die Länge der übertragenen Daten an. Dieser Wert wird durch CICS bei der Eingabe festgelegt. Er ist optional; falls er nicht angegeben wird, werden Sie nicht über die tatsächliche Länge der empfangenen Daten benachrichtigt.

TOKEN(*datenbereich*)

Gibt ein 8 Zeichen umfassendes und von CICS zugeordnetes Token für die Kennzeichnung eines Berichtes an.

Einschränkung: Die Option RESP oder NOHANDLE muss im EXEC CICS-Befehl **SPOOLREAD** angegeben werden.

Bedingungen

Anmerkung: Es gibt keine Standardaktionen.

85 ALLOCERR

Tritt in einer der folgenden Situationen auf:

- Die dynamische Zuordnung hat eine Anforderung zum Zuordnen einer Eingabedatei zurückgewiesen.

Das Feld RESP2 gibt den Antwortcode für die dynamische Zuordnung an, der diesen Fehler bezeichnet. Die ersten beiden Zeichen sind der Informationsursachencode (S99INFO) und die zweiten zwei Zeichen sind der Fehlerursachencode (S99ERROR); eine Definition finden Sie in der Veröffentlichung [z/OS MVS Programming: Authorized Assembler Services Guide](#).

20 ENDFILE

Tritt in einer der folgenden Situationen auf:

- Alle Daten für die gelesene aktuelle Spooldatei wurden abgerufen. Als Nächstes sollten Sie so bald wie möglich einen Befehl SPOOLCLOSE ausgeben, um die Sperre für den JES-Einzelthread freizugeben und die Verarbeitung der aktuellen SYSOUT-Datei zu beenden.

21 ILLOGIC

RESP2-Werte:

3

Es wurde ein ungültiger Wert für CLASS angegeben.

16 INVREQ

RESP2-Werte:

4

Nicht unterstützte Sprache.

8

Nicht unterstützte Funktion.

12

Es fand ein Leseversuch nach dem Dateiende statt.

24

Der Wert für INTO fehlt.

40

Die Subsystemschnittstelle ist bereits aktiviert.

Anmerkung: Die Fehler 1024 und höher sind interne Fehler, die nicht auftreten sollten. Falls einer dieser Fehlercodes zurückgegeben wird, wenden Sie sich an Ihr zuständiges IBM Support Center.

22 LENGERR

Tritt in einer der folgenden Situationen auf:

- Sie haben nicht genügend Pufferspeicher für das Lesen Ihres Datensatzes angegeben oder Sie haben mehr als die maximal zulässige Puffergröße von 32760 Byte angefordert (dies ist die Größe eines CICS-Puffers, der zum Lesen eines Datensatzes verwendet wird). Falls der Pufferspeicher zu klein ist, empfängt er so viele Daten wie möglich. Wie groß das abgeschnittene Datenvolumen ist, wird dann im Feld RESP2 angegeben. Wenn der Operand TOFLength angegeben ist, wird in diesem Feld die tatsächliche Länge des Datensatzes angegeben.

Anmerkung: Im Fall eines Längenfehlers aufgrund eines zu kleinen Pufferspeichers wird der nächste Datensatz erst dann gelesen, wenn der Fehler behoben und der aktuelle Datensatz erneut gelesen wurde.

Im Feld RESP2 ist die Größe der abgeschnittenen Daten bzw. Null angegeben, falls das Feld MAXFLength einen größeren Wert als die maximal zulässige Puffergröße von 32760 Byte besitzt.

80 NOSPOOL

RESP2-Werte:

4

Es ist kein Subsystem vorhanden.

8

Die Schnittstelle wird inaktiviert; CICS wird in den Wartemodus versetzt.

12

Die Schnittstelle wurde gestoppt.

42 NOSTG

Tritt in einer der folgenden Situationen auf:

- Ein Befehl GETMAIN ist innerhalb der JES-Schnittstellensubtask (DFHSPSS) fehlgeschlagen.
Das Feld RESP2 gibt den Rückgabecode 15 für das GETMAIN-Register zurück.

13 NOTFND

RESP2-Werte:

4

Es wurden keine Dateien für den Abruf durch den angegebenen externen Ausgabeprogrammen gefunden.

19 NOTOPEN

RESP2-Werte:

8

Die Datei wurde nicht geöffnet.

12

Es wurde versucht, eine Ausgabedatei zu lesen.

1024

Beim Öffnen des Makros OPEN für die Subtask trat ein Fehler auf.

88 SPOLBUSY

RESP2-Werte:

4

Die Schnittstelle wird bereits von einer anderen Task verwendet.

8

Die Schnittstelle wird bereits von der aktuellen Task verwendet.

Tritt außerdem in einer der folgenden Situationen auf (RESP2 nicht festgelegt):

- Der Einzelthread für JES/Eingabe in der JES-Schnittstelle war nicht verfügbar.

89 SPOLERR

Tritt in einer der folgenden Situationen auf:

- Das Makro für die Subsystemschnittstelle (IEFSSREQ) ist fehlgeschlagen. Es wurde kein Eingabedateiname ausgewählt.

Das Feld RESP2 gibt den Antwortcode IEFSSREQ an.

86 STRELERR

Tritt in einer der folgenden Situationen auf:

- Ein Befehl FREEMAIN ist innerhalb der JES-Schnittstellensubtask (DFHPSPSS) fehlgeschlagen.

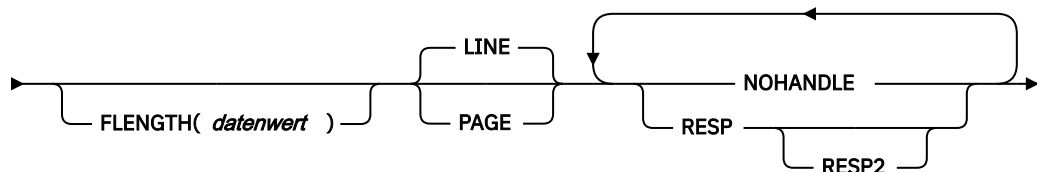
Das Feld RESP2 gibt den Rückgabecode 15 für das FREEMAIN-Register zurück.

SPOOLWRITE

Schreibt Daten in einen Spoolbericht.

SPOOLWRITE

➡ SPOOLWRITE — TOKEN(*datenbereich*) — FROM(*datenbereich*) —>



Bedingungen: ALLOCERR, INVREQ, LENGERR, NOSPOOL, NOSTG, NOTAUTH, NOTOPEN, SPOLBUSY, SPOLERR, STRELERR

Beschreibung

Der Befehl **SPOOLWRITE** schreibt Daten in einen Spoolbericht.

Optionen

FLENGTH(datenwert)

Gibt die Vollwort-Binärzahlvariable an, die auf die Länge der übertragenen Daten gesetzt werden soll. Dieser Wert wird durch den Benutzer bei der Ausgabe festgelegt. Er ist optional; falls er nicht angegeben ist, verwendet CICS die Länge des Datenbereichs.

FROM(datenbereich)

Gibt den Datenbereich an, aus dem die Daten mit variabler Länge entnommen werden sollen. Die Daten selbst werden in keiner Weise durch CICS geändert. FROM ist ein Senderfeld.

LINE|PAGE

Gibt das Format der zu sendenden Daten an. Die Standardaktion ist LINE.

Die Option PAGE muss verwendet werden, damit Informationen korrekt für AFP-Seitendruckeinheiten formatiert werden. Falls ein Kunde Daten mit dem Modustyp MIXED erstellt, also Datensätze des Typs LINE und Datensätze mit dem Seitenmodus X'5A' (AFPDS oder MODCA), muss der Operand LINE bzw. PAGE mit dem Typ des Datensatzes übereinstimmen, der in die Spoolausgabe geschrieben wird.

TOKEN(datenbereich)

Gibt ein 8 Zeichen umfassendes und von CICS zugeordnetes Token für die Kennzeichnung eines Berichtes an. Beim Befehl SPOOLOPEN ist dieses Feld ein Empfängerfeld; bei allen anderen Befehlen ist es ein Senderfeld.

Einschränkung: Die Option RESP oder NOHANDLE muss im EXEC CICS-Befehl **SPOOLWRITE** angegeben werden.

Bedingungen

Anmerkung: Es gibt keine Standardaktionen.

85 ALLOCERR

Tritt in einer der folgenden Situationen auf:

- Die dynamische Zuordnung hat eine Anforderung zum Zuordnen einer Eingabedatei zurückgewiesen.

Das Feld RESP2 gibt den Antwortcode für die dynamische Zuordnung an, der diesen Fehler bezeichnet. Die ersten beiden Zeichen sind der Informationsursachencode (S99INFO) und die zweiten zwei Zeichen sind der Fehlerursachencode (S99ERROR); eine Definition finden Sie in der Veröffentlichung [z/OS MVS Programming: Authorized Assembler Services Guide](#).

16 INVREQ

RESP2-Werte:

4

Nicht unterstützte Sprache.

8

Nicht unterstützte Funktion.

28

Der Wert für FROM fehlt.

40

Die Subsystemschnittstelle ist bereits aktiviert.

Anmerkung: Die Fehler 1024 und höher sind interne Fehler, die nicht auftreten sollten. Falls einer dieser Fehlercodes zurückgegeben wird, wenden Sie sich an Ihr zuständiges IBM Support Center.

22 LENGERR

Tritt in einer der folgenden Situationen auf:

- Der im Parameter FLENGTH für einen Befehl SPOOLWRITE angegebene Wert liegt nicht im gültigen Bereich zwischen 1 und dem angegebenen Wert des Parameters RECORDLENGTH oder dem im Datensatz für SPOOLOPEN festgelegten Standardwert. Falls der Pufferspeicher zu klein ist, empfängt er so viele Daten wie möglich.

Das Feld RESP2 enthält die Differenz zwischen den Werten von FLENGTH und RECORDLENGTH bzw. Null, falls der Wert von FLENGTH negativ oder größer als 32760 ist.

80 NOSPOOL

RESP2-Werte:

4

Es ist kein Subsystem vorhanden.

8

Die Schnittstelle wird inaktiviert; CICS wird in den Wartemodus versetzt.

12

Die Schnittstelle wurde gestoppt.

42 NOSTG

Tritt in einer der folgenden Situationen auf:

- Ein Befehl GETMAIN ist innerhalb der JES-Schnittstellensubtask (DFHPSPSS) fehlgeschlagen.

Das Feld RESP2 gibt den Rückgabecode 15 für das GETMAIN-Register zurück.

70 NOTAUTH

RESP2-Werte:

1

Eine Ersatzsicherheitsprüfung ist fehlgeschlagen. Weitere Informationen finden Sie unter [Sicherheit für die Übergabe eines JCL-Jobs an den internen Leser](#).

Standardaktion: Die Task wird abnormal beendet.

19 NOTOPEN

RESP2-Werte:

8

Der Spoolbericht wurde nicht geöffnet.

16

Es wurde versucht, eine Eingabedatei zu schreiben.

1024

Beim Öffnen des Makros OPEN für die Subtask trat ein Fehler auf.

88 SPOLBUSY

RESP2-Werte:

4

Die Schnittstelle wird bereits von einer anderen Task verwendet.

8

Die Schnittstelle wird bereits von der aktuellen Task verwendet.

Diese Bedingung (RESP2 nicht festgelegt) tritt auch in der folgenden Situation auf:

- Der Einzelthread für JES/Eingabe in der JES-Schnittstelle war nicht verfügbar.

89 SPOLERR

Tritt in der folgenden Situation auf:

- Das Makro für die Subsystemschnittstelle (IEFSSREQ) ist fehlgeschlagen. Es wurde kein Eingabedateiname ausgewählt.

Das Feld RESP2 gibt den Antwortcode IEFSSREQ an.

86 STRELERR

Tritt in der folgenden Situation auf:

- Ein Befehl FREEMAIN ist innerhalb der JES-Schnittstellensubtask (DFHPSPSS) fehlgeschlagen.

Das Feld RESP2 gibt den Rückgabecode 15 für das FREEMAIN-Register zurück.

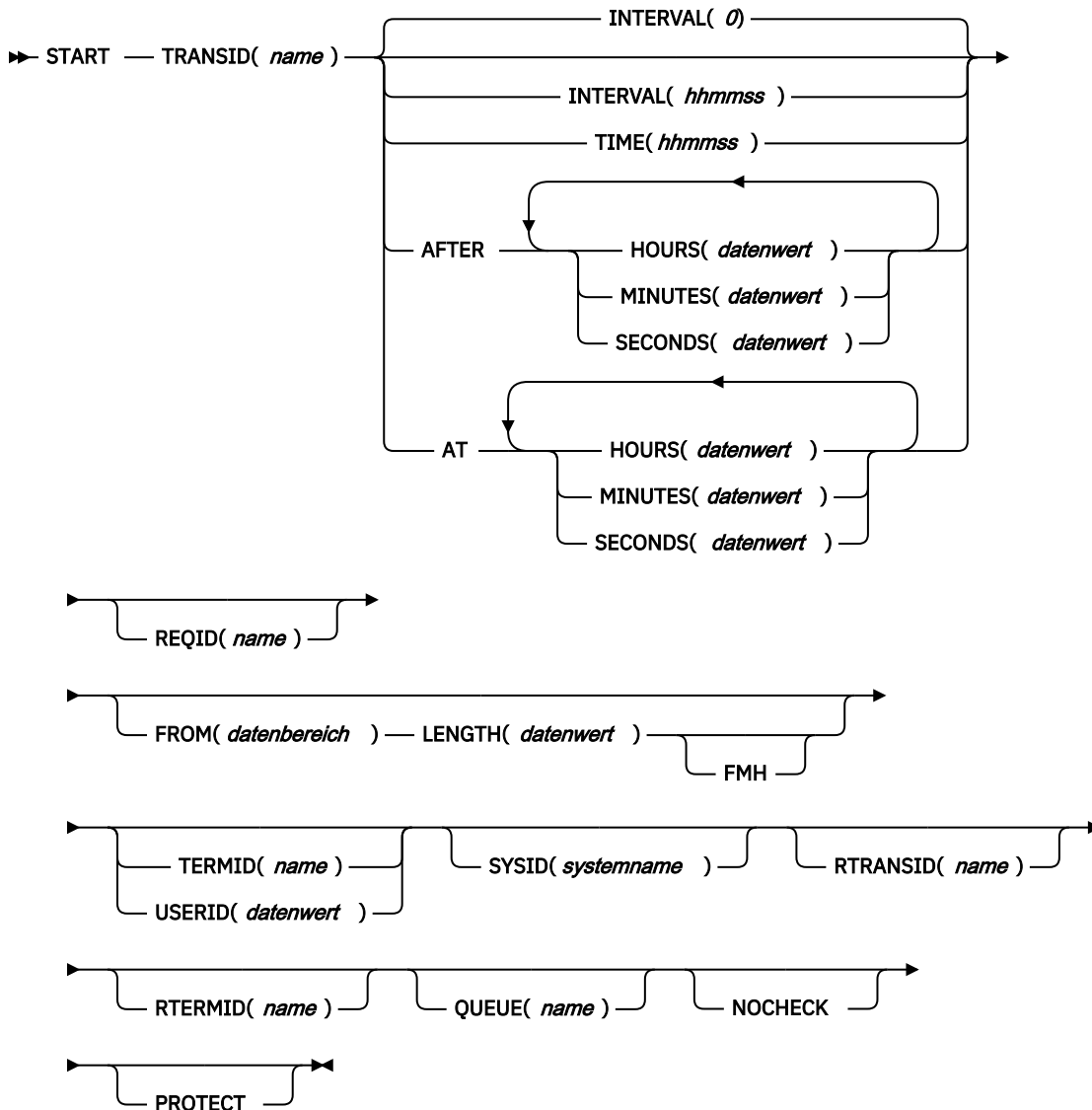
START

Startet eine Task zu einer angegebenen Uhrzeit.

Siehe auch:

- [START ATTACH](#)
- [START BREXIT](#)
- [START CHANNEL](#)

START



Bedingungen: INVREQ, IOERR, ISCINVREQ, LENGERR, NOTAUTH, RESUNAVAIL, SYSIDERR, TERMIDERR, TRANSIDERR, USERIDERR

Anmerkung: Dynamisches Transaktionsrouting: Falls ein Befehl **START** später durch eine andere Task abgebrochen wird oder die gestartete Transaktion den Befehl **RETRIEVE** mit der Option **WAIT** verwendet, können transaktionsübergreifende Affinitäten entstehen, die die Nutzung des dynamischen Transaktionsroutings beeinträchtigen. Weitere Informationen finden Sie unter [Affinität](#).

Beschreibung

Der Befehl **START** startet zum angegebenen Zeitpunkt eine Task auf einem lokalen oder fernen System. Der Zeitpunkt wird durch die Option **INTERVAL**, **AFTER**, **AT** oder **TIME** angegeben. Lesen Sie in diesem Zusammenhang auch den Abschnitt über Ablaufzeiten unter [Intervallsteuerung](#).

Die startende Task kann Daten an die gestartete Task übergeben. Die startende Task kann außerdem ein Terminal angeben, das von der gestarteten Task als ihre Hauptfunktion verwendet werden soll.

Der Standardwert ist **INTERVAL(0)**; bei C lautet der Standardwert jedoch **AFTER HOURS(0) MINUTES(0) SECONDS(0)**.

CEDF ist eine Ausnahme für den Befehl **START** und als Name für **TRANSID** nicht gültig. Versuchen Sie daher nicht, **CEDF** auf diese Weise zu starten.

Wichtig: Es gibt eine 6-Stunden-Regel: Falls die Operation **START** zu einem beliebigen Zeitpunkt innerhalb von 6 Stunden nach der Zeit ausgelöst wird, die im Befehl **START** angegeben ist, wird sie sofort ausgeführt. Dies findet unabhängig davon statt, ob die vorangegangenen 6 Stunden einen Datumswechsel beinhalteten. Dies soll an einigen Beispielen veranschaulicht werden:

- **EXEC CICS START TIME(123000)**

Dieser Befehl, ausgegeben um 5.00 Uhr oder 7.00 Uhr am Montag, läuft um 12.30 Uhr desselben Tages ab.

- **EXEC CICS START TIME(020000)**

Dieser Befehl, ausgegeben um 5.00 Uhr oder 7.00 Uhr am Montag, läuft sofort ab, weil der angegebene Zeitpunkt innerhalb der vorangegangenen 6 Stunden liegt.

Weitere Informationen finden Sie unter [Ablaufzeiten](#).

Mit den Optionen **RTRANSID**, **RTERMID** und **QUEUE** können Sie weitere Daten an die gestartete Task übergeben. Diese Optionen können beliebige Datenwerte enthalten, deren Bedeutungen von den Angaben in der gestarteten und der startenden Task abhängig sind. Eine mögliche Verwendung dieser Optionen besteht in der folgenden Situation. Eine Task kann eine zweite Task starten und hierbei einen Transaktionsnamen und einen Terminalnamen übergeben, die verwendet werden sollen, wenn die zweite Task eine dritte Task startet. Die erste Task kann außerdem den Namen einer Warteschlange übergeben, auf die durch die zweite Task zugegriffen werden soll.

Bei Verwendung einer **IPIC**-Verbindung beträgt die maximale Länge des mit **FROM** angegebenen Datenbereichs 32500 Byte. Dieser Grenzwert ermöglicht den Datenbereich für **FROM** mit einer Größe von 32500 Byte sowie Speicherbereich für Header.

Ein Befehl **START** mit angegebener Option **TERMID** gibt nicht den Ursprungsdatensatz (Origin Data Record, **ODR**) weiter, weshalb Tasks immer an einem neuen Ausgangspunkt gestartet werden.

Die folgenden Bedingungen müssen erfüllt sein, damit die auszuführende Transaktion gestartet werden kann:

- Das angegebene Intervall muss abgelaufen sein oder die angegebene Ablaufzeit muss erreicht worden sein. Weitere Informationen finden Sie unter [Intervallsteuerung](#). Geben Sie die Optionen **INTERVAL** oder **AFTER** an, wenn eine Transaktion auf einem fernen System ausgeführt werden soll. Hierdurch vermeiden Sie Komplikationen, wenn sich das lokale und das ferne System in unterschiedlichen Zeitzonen befinden.
- Falls Sie die Option **TERMID** angeben, muss das benannte Terminal vorhanden und verfügbar sein. Wenn das benannte Terminal nicht vorhanden ist, wird der Start verworfen, sobald das Zeitintervall abläuft.
- Falls Sie die Option **PROTECT** angeben, muss die startende Task einen erfolgreichen Synchronisationspunkt annehmen. In Kombination mit Erweiterungen an Systemtabellen verringert diese Option die Gefahr, dass Daten aufgrund des Fehlschlagens einer startenden Task verloren gehen oder doppelt vorhanden sind.
- Falls die Transaktion auf einem fernen System ausgeführt werden soll, muss das Format der Daten identisch mit den Daten auf dem lokalen System deklariert sein. Verwenden Sie die **RDO**-Optionen **DA**-

TASTREAM und RECORDFORMAT. Bei Übertragungen von CICS an CICS sind diese Optionen immer die Standardwerte. Achten Sie bei Übertragungen von CICS an IMS/VS darauf, die richtigen Werte anzugeben.

Die Ausführung eines Befehls START, der eine Transaktion im lokalen System benennt, bricht alle ausstehenden Befehle POST ab, die durch die startende Task ausgeführt werden.

Sie können Befehle START in die Warteschlange stellen, indem Sie die Option LOCALQ in der RDO-Resourcendefinition TRANSACTION angeben (dies ist unter [TRANSACTION-Attribute](#) beschrieben).

Daten durch Intervallsteuerung übergeben

Falls Daten (mithilfe der Option FROM) durch die Intervallsteuerung übergeben werden sollen, werden sie in eine Warteschlange für temporären Speicher eingereiht. Verwenden Sie die Option REQID, um den Namen der Warteschlange für temporären Speicher anzugeben, die verwendet werden soll. Diese ID kann (in Bezug auf den temporären Speicher) wiederherstellbar sein oder auch nicht. Weitere Informationen zum Definieren von wiederherstellbaren Warteschlangen für temporären Speicher finden Sie unter [TSMO-DEL-Ressourcen](#).

Wenn Sie ebenfalls die Option PROTECT angeben, müssen Sie die Warteschlange für temporären Speicher, die durch die Option REQID angegeben wird, als wiederherstellbar definieren. Falls Sie die Option PROTECT nicht angeben, definieren Sie die Warteschlange für temporären Speicher nicht als wiederherstellbar. Falls diese Regeln nicht befolgt werden, können unvorhersehbare Ergebnisse auftreten. Weitere Informationen finden Sie unter [Wiederherstellung von START-Anforderungen](#).

Falls Sie die Option FROM und nicht die Option REQID angeben, wird standardmäßig eine Warteschlange für temporären Speicher mit dem Präfix 'DF' verwendet. Hierbei gelten dieselben Regeln wie oben beschrieben; geben Sie die Option PROTECT nur dann an, wenn Sie die Warteschlange für temporären Speicher mit dem Präfix 'DF' als wiederherstellbar definieren.

Ein Befehl START mit der Option REQID, der von einer Task ausgegeben wird, die selbst durch einen Befehl START mit demselben Wert für REQID initialisiert wurde, gibt einen Abbruch des Typs AEIQ (Bedingung IOERR) zurück, falls die mit FROM angegebenen Daten für die Task nicht durch einen Befehl RETRIEVE gelesen worden sind.

Diesen Fehler empfangen Sie ebenfalls, wenn mehrere Befehle START mit demselben Wert für REQID durch eine oder mehrere Tasks in demselben CICS-System ausgegeben werden. CICS TS-Regionen weisen Befehle START, die einen identischen Wert für REQID angeben, immer mit einer Bedingung IOERR zurück.

Gestartete Tasks ohne Daten werden ohne eine Funktionsadresse ausgeführt. Gestartete Tasks mit Daten werden mit einer Funktionsadresse eines Intervallsteuerungselementes (Interval Control Element, ICE) ausgeführt, bis die Daten abgerufen wurden.

Falls eine erweiterte Identitätskontextreferenz (Extended Identity Context Reference, ICX) verwendet wird, wird sie über Neustarts hinweg gespeichert. Wenn anschließend die Startanforderung aufgerufen wird, wird die ICX gelöscht.

Fehlerprüfung und Leistungsaspekte

Die Option NOCHECK gibt an, dass keine Antwort (auf die Ausführung des Befehls START) von der startenden Transaktion erwartet wird. Bei Befehlen START, die zu startende Tasks auf einem lokalen System benennen, werden Fehlerbedingungen zurückgegeben; für Tasks, die auf einem fernen System gestartet werden sollen, erfolgt keine Rückgabe von Fehlerbedingungen. Die Option NOCHECK ermöglicht CICS eine Verbesserung des Leistungsverhaltens, wenn der Befehl START einem fernen System zugestellt werden muss; sie ist darüber hinaus eine Voraussetzung, falls die Zustellung des Befehls START in die Warteschlange eingereiht wird, weil der Aufbau von Verbindungen zum fernen System noch aussteht.

Tasks ohne Terminals starten

Falls die zu startende Task keinem Terminal zugeordnet ist, startet jeder Befehl START eine separate Task und zwar unabhängig davon, ob an die gestartete Task Daten übergeben werden. Die folgenden Bei-

spiele zeigen, wie eine angegebene Task, die keinem Terminal zugeordnet ist, in einer Stunde gestartet wird:

```
EXEC CICS START
  TRANSID('TRNL')
  INTERVAL(10000)
  REQID('NONGL')
:
EXEC CICS START
  TRANSID('TRNL')
  AFTER HOURS(1)
  REQID('NONGL')
:
```

Tasks mit Terminals, jedoch ohne Daten starten

Falls mehrere Befehle START, die jeweils dieselbe Transaktion und dasselbe Terminal angeben, zu demselben Zeitpunkt oder vor Verfügbarkeit des Terminals ablaufen, wird eine einzige Task gestartet.

Die folgenden Beispiele zeigen, wie die Initialisierung einer Task angefordert wird, die einem Terminal zugeordnet ist. Da in diesen Beispielen keine Anforderungs-ID angegeben ist, ordnet CICS eine ID zu und gibt sie im Feld EIBREQID des EXEC-Schnittstellenblocks an das Anwendungsprogramm zurück.

```
EXEC CICS START
  TRANSID('TRN1')
  TIME(185000)
  TERMID('STA5')
:
EXEC CICS START
  TRANSID('TRN1')
  AT HOURS(18) MINUTES(50)
  TERMID('STA5')
:
```

Tasks mit Terminals und Daten starten

Daten werden an eine gestartete Task übergeben, wenn eine oder mehrere der Optionen FROM, RTRANSID, RTERMID und QUEUE angegeben sind. Auf solche Daten greift die gestartete Task durch einen Befehl RETRIEVE zu.

An eine neue Task können viele Datensätze übergeben werden; hierzu werden mehrere Befehle START ausgegeben, die jeweils dieselbe Transaktion und dasselbe Terminal angeben.

Die Ausführung des ersten Befehls START bewirkt schließlich das Starten der neuen Task und ermöglicht der Task den Abruf der Daten, die im Befehl angegeben sind. Die neue Task ist außerdem in der Lage, angegebene Daten in nachfolgenden Befehlen START abzurufen, die vor der Beendigung der neuen Task ablaufen. Falls die Transaktion als wiederanlauffähig definiert wurde (mithilfe der RDO-Option RESTART(YES)) und solche Daten vor der Beendigung der neuen Task nicht abgerufen wurden, wird eine weitere neue Task gestartet, die die ausstehenden Daten abrufen kann.

Falls die Transaktion abgebrochen wird und nicht als wiederanlauffähig definiert war, wird keine neue Task initialisiert und die Daten werden gelöscht.

Die folgenden Beispiele zeigen, wie eine Task gestartet werden kann, die einem Terminal zugeordnet ist, und wie Daten an die Task übergeben werden können:

```

EXEC CICS START
  TRANSID('TRN2')
  TIME(173000)
  TERMID('STA3')
  REQID(DATAREC)
  FROM(DATAFLD)
  LENGTH(100)
:
EXEC CICS START
  TRANSID('TRN2')
  AT HOURS(17) MINUTES(30)
  TERMID('STA3')
  REQID(DATAREC)
  FROM(DATAFLD)
  LENGTH(100)
:

```

Bei Verwendung der Programmiersprache C empfiehlt sich die Verwendung der Optionen AFTER/AT mit den Optionen HOURS, MINUTES und SECONDS, da C den Datentyp für gepackte Dezimalzahlen nicht unterstützt. Sie können die Option INTERVAL oder TIME zwar verwenden, aber falls es sich bei dem angegebenen Wert *nicht* um eine ganzzahlige Konstante handelt, muss die Anwendung sicherstellen, dass der an CICS übergebene Wert ein gepacktes Dezimalformat besitzt.

Durch Befehle START gestartete weitergeleitete Transaktionen

Einige durch eine Untergruppe von Befehlen START gestartete Transaktionen können dynamisch an eine ferne Region weitergeleitet werden. Allgemeine Angaben über das dynamische Transaktionsrouting und spezielle Informationen dazu, welche durch Befehle START gestarteten Transaktionen für das dynamische Routing infrage kommen, finden Sie unter [Mit START-Befehlen aufgerufene Transaktionen weiterleiten](#).

Bei aktiver Sicherheit wird die Benutzer-ID, die der Transaktion in der fernen Region zugeordnet ist, gemäß dem Wert festgelegt, der im Parameter **ATTACHSEC** der Verbindungsdefinition in der fernen Region angegeben ist. Weitere Informationen zu der Sicherheitsprüfung, die in der Anwendungsverwaltungsregion (AOR) erfolgt, enthalten die folgenden Abschnitte für das jeweils verwendete Kommunikationsverfahren:

- [Sicherheitsprüfung in der Anwendungsverwaltungsregion \(AOR\) mit LU6.2](#)
- [Sicherheitsprüfung in der Anwendungsverwaltungsregion \(AOR\) mit LU6.1](#)
- [Sicherheitsprüfung in der Anwendungsverwaltungsregion mit IPIC](#)
- [Sicherheitsprüfung in der Anwendungsverwaltungsregion \(AOR\) mit MRO](#)

Anmerkung: Die Verwendung der Identitätsweitergabe mit einem Befehl START, der über eine LU61- oder LU62-Verbindung an eine ferne Region zugestellt wird, ist nicht möglich. Bei diesem Verbindungstyp wird die ICRX nicht zugestellt und die Identitätsinformationen gehen verloren. Auch bei verzögert dynamisch weitergegebenen Befehlen START ist die Identitätsweitergabe nicht verfügbar.

Fehler bei Befehlen START ohne Ausnahmebedingungen

Unter den folgenden Umständen wird ein Befehl START zwar ohne Fehler ausgeführt, die gestartete Task findet jedoch zu keinem Zeitpunkt statt:

- Die Transaktion oder ihr Arbeitsgang ist zu dem Zeitpunkt nicht verfügbar, an dem CICS versucht, die Task zu erstellen.
- Der Befehl START gibt ein Terminal und eine Ablaufzeit an, das Terminal ist jedoch zur Ablaufzeit nicht definiert (und kann durch die Exits XICTENF oder XALTENF nicht gefunden werden).
- Der Befehl START gibt ein Terminal an, das zu dem Zeitpunkt, an dem CICS versucht, die Task zu erstellen, nicht definiert ist (und nicht durch die Exits XICTENF oder XALTENF gefunden werden kann).

Diese Risiken entstehen durch die Verzögerung zwischen der Ausführung des Befehls START und dem Zeitpunkt der Taskerstellung. Selbst wenn der Befehl START sofort ausgeführt wird, ist es möglich, dass CICS die Erstellung der Task verzögert, weil entweder das angeforderte Terminal nicht verfügbar ist oder andere Systembedingungen vorliegen.

Mit Befehlen INQUIRE können Sie sicherstellen, dass die Transaktion und das Programm zum Zeitpunkt des Befehls START aktiviert sind. Es kann allerdings sein, dass die Verfügbarkeit der Transaktion oder des Programms vor der Taskerstellung endet.

Falls das angeforderte Terminal zum Zeitpunkt des Befehls START nicht vorhanden ist, erhalten Sie eine Bedingung TERMIDERR. Wird das Terminal jedoch später gelöscht, was beim Abmelden des Benutzers passiert, wird Ihre Anforderung START zusammen mit der Terminaldefinition verworfen.

Weitergabe des Anwendungskontextes

Die Weitergabe des Anwendungskontextes ist ein Prozess, bei dem Anwendungskontextdaten von einer Task, die einen Befehl START ausgibt, an eine Task weitergegeben werden, die die gestartete Transaktion ausführt. Die Daten werden zur Überwachung und Messung der Ressourcennutzung durch die Anwendung verwendet. Zwischen Anwendungen wird sowohl der anfängliche als auch der aktuelle Anwendungskontext übergeben. Bei den folgenden Befehlen START werden Anwendungskontextdaten *nicht* weitergegeben:

- Lokale EXEC CICS-Befehle **START**, die aus einem Auslöser für transiente Daten (TD) generiert werden.
- Lokale asynchrone EXEC CICS-Befehle **START**, die von einem Adapter für die Ereignisverarbeitung (EP) empfangen werden.
- EXEC CICS-Befehle **START**, die einen Wert für den Parameter **TERMID** oder **USERID** angeben.

Weitere Informationen finden Sie unter [Anwendungskontext](#).

Optionen

AFTER

Gibt das Zeitintervall an, das verstreichen soll, bevor die neue Task gestartet wird.

Für die Angabe der Zeit unter den Optionen AFTER und AT gibt es zwei Möglichkeiten:

1. Kombination von mindestens zwei der Optionen HOURS(0 - 99), MINUTES(0 - 59) und SECONDS(0 - 59). HOURS(1) SECONDS(3) bedeutet eine Stunde und drei Sekunden (der Wert für die Minuten nimmt standardmäßig Null an).
2. Durch einen der Werte HOURS(0 - 99), MINUTES(0 - 5999) oder SECONDS(0 - 59999 999). Die Angabe HOURS(1) bedeutet 1 Stunde. MINUTES(62) bedeutet eine Stunde und zwei Minuten. Die Angabe SECONDS(3723) bedeutet 1 Stunde, 2 Minuten und 3 Sekunden.

AT

Gibt den Zeitpunkt an, zu dem die neue Task gestartet werden soll. Informationen zur Angabe der Zeit enthält die Beschreibung der Option AFTER.

FMH

Gibt an, dass die Benutzerdaten, die an die gestartete Task übergeben werden sollen, Funktionsverwaltungsholder enthalten. Die Option FMH ist für Terminals mit LU-Typ 2 oder LU-Typ 3 nicht gültig.

FROM(datenbereich)

Gibt die Daten an, die für eine Task gespeichert werden sollen, deren Start zu einem künftigen Zeitpunkt stattfinden soll.

HOURS(datenwert)

Gibt einen Vollwort-Binärwert zwischen 0 und 99 an. Die Option HOURS ist eine Unteroption der Optionen AFTER und AT. Informationen zur Verwendung und zur Bedeutung finden Sie unter der Option AFTER.

INTERVAL(hhmmss)

Gibt die Ablaufzeit als Intervall an, das ab dem Zeitpunkt verstreichen soll, an dem der Befehl START ausgegeben wird. Die Werte für *mm* und *ss* liegen jeweils zwischen 0 und 59. Die angegebene Zeit wird bei der Ausführung des Befehls durch CICS zur aktuellen Systemzeit addiert, um die Ablaufzeit zu berechnen.

LENGTH(*datenwert*)

Gibt als Halbwort-Binärdatenwert die Länge der Daten an, die für die neue Task gespeichert werden sollen. Informationen zu Grenzwerten beim Festlegen der Option LENGTH finden Sie unter „LENGTH-Optionen in CICS-Befehlen“ auf Seite 10.

MINUTES(*datenwert*)

Gibt als Vollwort-Binärwert die Anzahl der Minuten an, die für die Option AFTER oder AT verwendet werden soll. Der Wert muss zwischen 0 und 59 liegen, falls HOURS oder SECONDS ebenfalls angegeben ist; ansonsten muss er zwischen 0 und 5999 liegen. Die Option MINUTES ist eine Unteroption der Optionen AFTER und AT. Informationen zur Verwendung und zur Bedeutung finden Sie unter der Option AFTER.

NOCHECK

Gibt an, dass CICS die Leistung des Befehls START für ein fernes System verbessert, indem eine geringere Fehlerprüfung und eine etwas verringerte Funktionalität bereitgestellt werden. Weitere Informationen finden Sie unter Leistung systemübergreifender START-Anforderungen verbessern.

PROTECT

Gibt an, dass die neue Task erst dann gestartet wird, wenn die startende Task einen Synchronisationspunkt erreicht hat. Falls die startende Task vor dem Erreichen des Synchronisationspunkts abgebrochen wird, wird die Anforderung zum Starten der neuen Task abgebrochen. Wenn ebenfalls die Option REQID angegeben ist, muss die Anforderungs-ID ein Name sein, der als im temporären Speicher wiederherstellbar definiert ist. Handelt es sich bei der gestarteten Transaktion um eine ferne Transaktion, gibt die Option PROTECT an, dass sie erst geplant werden darf, wenn die lokale Transaktion einen Synchronisationspunkt erfolgreich abgeschlossen hat. Weitere Informationen zur Option PROTECT bei fernen Transaktionen finden Sie unter Leistung systemübergreifender START-Anforderungen verbessern.

QUEUE(*name*)

Gibt einen Namen (1 bis 8 Zeichen) an, der an die gestartete Task übergeben wird. Falls dieser Name eine Warteschlange für temporären Speicher darstellt, muss die Warteschlange für die gestartete Task lokal sein. Der Inhalt der Warteschlange wird nicht übergeben.

Wenn Sie ebenfalls die Option REQID angeben, stellen Sie sicher, dass die in den Optionen REQID und QUEUE angegebenen Namen nicht identisch sind.

REQID(*name*)

Gibt zur Kennzeichnung eines Befehls einen Namen (1 bis 8 Zeichen) an, der eindeutig sein muss. Diese Option können Sie verwenden, wenn eine weitere Task bereitgestellt werden soll, die in der Lage ist, einen nicht abgelaufenen Befehl abubrechen.

Falls Sie diese Option nicht angeben, generiert CICS im Feld EIBREQID des EXEC-Schnittstellenblocks eine eindeutige Anforderungs-ID, sofern Sie nicht die Option NOCHECK angegeben haben. In diesem Fall wird das Feld EIBREQID auf Nullen gesetzt und kann später nicht zum Abbrechen des Befehls START verwendet werden.

Wenn Sie eine der Datenoptionen einbeziehen (FROM, RTERMID, RTRANSID oder QUEUE), werden die Daten in einer TS-Warteschlange gespeichert; hierbei wird als Kennung der in REQID angegebene (oder durch CICS generierte) Namen verwendet. Der so gekennzeichnete Warteschlangendatensatz muss für das CICS-System, auf dem der Befehl START verarbeitet wird, lokal sein. Der Befehl START wird auf dem System verarbeitet, das durch die Option SYSID angegeben ist, bzw. auf dem System, das der Option TRANSID zugeordnet ist, falls die Option SYSID nicht angegeben wurde.

RTERMID(*name*)

Gibt einen Wert (1 - 4 Zeichen) an, beispielsweise einen Terminalnamen, der abgerufen werden kann, wenn die in der Option TRANSID des Befehls START angegebene Transaktion gestartet wird.

Nach dem Abruf kann der Wert in der Option TERMID eines nachfolgenden Befehls START verwendet werden.

RTRANSID(*name*)

Gibt einen Wert (1 - 4 Zeichen) an, beispielsweise einen Transaktionsnamen, der abgerufen werden kann, wenn die in der Option TRANSID des Befehls START angegebene Transaktion gestartet wird.

Nach dem Abruf kann der Wert in der Option TRANSID eines nachfolgenden Befehls START verwendet werden.

SECONDS(*datenwert*)

Gibt einen Vollwort-Binärwert zwischen 0 und 59 (wenn ebenfalls die Optionen HOURS oder MINUTES angegeben sind) bzw. zwischen 0 und 359999 an (wenn allein die Option SECONDS angegeben ist). Die Option SECONDS ist eine Unteroption der Optionen AFTER und AT. Informationen zu ihrer Verwendung und Bedeutung enthält die Beschreibung der Option AFTER.

SYSID(*systemname*)

Gibt den Namen des Systems an, an das die Anforderung gerichtet ist.

TERMID(*name*)

Gibt die symbolische ID (1 bis 4 alphanumerische Zeichen) der Hauptfunktion an, die einer Transaktion zugeordnet ist, die infolge eines Befehls START gestartet werden soll. Diese Hauptfunktion kann entweder ein Terminal (im Normalfall) oder eine APPC-Sitzung sein. Bei Angabe einer APPC-Sitzung wird anstelle einer Terminal-ID der Name der Verbindung (oder des Modussets) verwendet. Diese Option ist erforderlich, wenn die zu startende Transaktion mit einem Terminal kommunizieren muss; andernfalls sollte sie weggelassen werden.

Definieren Sie die Terminal-ID entweder als lokales oder als fernes Terminal in dem System, in dem der Befehl START ausgeführt werden soll, wenn der Start der Transaktion stattfindet.

Die Option TERMID wird in erfassten Daten zum vorherigen Hop verwendet. Weitere Informationen zur Verwendung der Option TERMID mit Daten zum vorherigen Hop finden Sie unter [Merkmale der Daten zum vorherigen Hop](#).

Verwenden Sie die Option TERMID nicht, wenn Sie die Identitätsweitergabe einsetzen.

TIME(*hhmmss*)

Gibt den Zeitpunkt an, zu dem eine neue Task gestartet wird.

Bei Verwendung der Programmiersprache C empfiehlt sich die Verwendung der Optionen AFTER/AT mit den Optionen HOURS, MINUTES und SECONDS, da C den Datentyp für gepackte Dezimalzahlen nicht unterstützt. Sie können die Option TIME zwar verwenden, aber falls es sich bei dem angegebenen Wert *nicht* um eine ganzzahlige Konstante handelt, muss die Anwendung sicherstellen, dass der an CICS übergebene Wert ein gepacktes Dezimalformat besitzt.

TRANSID(*name*)

Gibt die symbolische ID (1 bis 4 Zeichen) der Transaktion an, die durch eine infolge eines Befehls START gestarteten Task ausgeführt werden soll.

Falls Sie die Option SYSID angeben und ein fernes System benennen, wird davon ausgegangen, dass sich die Transaktion auf diesem System befindet, auf dem die Transaktion als fern definiert ist. Andernfalls wird mithilfe der Transaktionsressourcendefinition ermittelt, ob sich die Transaktion auf einem lokalen oder auf einem fernen System befindet.

Die Option TRANSID wird in erfassten Daten zum vorherigen Hop verwendet. Weitere Informationen finden Sie unter [Merkmale der Daten zum vorherigen Hop](#).

USERID(*datenwert*)

Gibt die Benutzer-ID an, unter deren Berechtigung die gestartete Transaktion ausgeführt werden soll, wenn die gestartete Transaktion keinem Terminal zugeordnet ist (die Option TERMID also nicht angegeben wurde). Diese Benutzer-ID wird als *userid1* bezeichnet.

Wenn Sie weder die Option TERMID noch die Option USERID angeben, verwendet CICS stattdessen die Benutzer-ID, unter der die Transaktion ausgeführt wird, die den Befehl START ausgibt. Diese Benutzer-ID wird als *userid2* bezeichnet.

Durch die Verwendung von entweder *userid1* oder *userid2* gewährleistet CICS, dass eine gestartete Transaktion immer unter einer gültigen Benutzer-ID ausgeführt wird, die für alle Ressourcen berechtigt sein muss, die von der gestarteten Transaktion referenziert werden.

CICS führt für *userid2* eine Ersatzsicherheitsprüfung durch, um zu verifizieren, dass dieser Benutzer für *userid1* berechtigt ist. Falls *userid2* nicht berechtigt ist, gibt CICS eine Bedingung NOTAUTH zurück.

Wenn Sie die Identitätsweitergabe verwenden und dem Sicherheitskontext Ihrer Task eine verteilte Benutzer-ID zugeordnet ist, werden diese Informationen nicht an Tasks weitergegeben, die mit der Option USERID gestartet werden.

Bedingungen

16 INVREQ

RESP2-Werte:

4

Der für HOURS bei den Optionen AFTER oder AT angegebene Wert bzw. der für *hh* bei der Option INTERVAL angegebene Wert liegt außerhalb des gültigen Bereichs.

5

Der für MINUTES bei den Optionen AFTER oder AT angegebene Wert bzw. der für *mm* bei der Option INTERVAL angegebene Wert liegt außerhalb des gültigen Bereichs.

6

Der für SECONDS bei den Optionen AFTER oder AT angegebene Wert bzw. der für *ss* bei der Option INTERVAL angegebene Wert liegt außerhalb des gültigen Bereichs.

17

Die durch die Operation START gestartete Transaktion ist nicht für einen Systemabschluss geeignet und für die CICS-Region wird gerade ein Systemabschluss durchgeführt.

18

Es wurde ein Wert für USERID angegeben und die CICS-Schnittstelle für den externen Sicherheitsmanager ist nicht initialisiert.

200

Der Befehl START wurde durch ein Serverprogramm für Distributed Program Link (DPL) ausgegeben und verwendete die Option TERMID, deren Wert mit der ID der systemübergreifenden Sitzung übereinstimmt. Wenn der Wert von TERMID mit dem Wert von EIBTRMID identisch ist (es sich bei der Hauptfunktion der ausgebenden Task also nicht um ein Terminal, sondern um eine Sitzung handelt), gehört der Befehl START TERMID nicht zu der für DPL-Serverprogramme verfügbaren Teilmenge der zulässigen Befehle.

400

Der Befehl START wurde zurückgewiesen, weil der Schwellenwert für eine AID-Systemregel überschritten wurde.

Die Bedingung INVREQ tritt außerdem in einer der folgenden Situationen auf (RESP2 nicht festgelegt):

- Der Befehl START ist für die Verarbeitung durch CICS nicht gültig.
- Die in der Option INTERVAL angegebenen Werte liegen außerhalb des gültigen Bereichs.

Standardaktion: Die Task wird abnormal beendet.

17 IOERR

Tritt in einer der folgenden Situationen auf:

- Während einer Startoperation trat ein E/A-Fehler auf.
- Eine Startoperation hat versucht, Daten in eine Warteschlange für temporären Speicher zu schreiben, als die Datei DFHTEMP voll war.
- Eine Startoperation verwendet für REQID einen Namen, der vorhanden ist. Diese Bedingung tritt nur dann auf, wenn auch die Option FROM verwendet wird.

Standardaktion: Die Task wird abnormal beendet.

54 ISCINVREQ

Tritt auf, wenn das ferne System einen Fehler angegeben hat, der keiner bekannten Bedingung entspricht.

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

Tritt auf, wenn der Wert für LENGTH nicht größer als Null ist.

Standardaktion: Die Task wird abnormal beendet.

70 NOTAUTH

RESP2-Werte:

7

Eine Ressourcensicherheitsprüfung ist für TRANSID (*name*) fehlgeschlagen.

9

Eine Sicherheitsprüfung des Ersatzbenutzers ist für USERID (*name*) fehlgeschlagen.

Die Sicherheitszugriffsfunktionen der Transaktion, die den Befehl ausgegeben hat, lassen eine Ausführung des Befehls mit dem Wert, der in der Option USERID angegeben ist, nicht zu. Die Sicherheitszugriffsfunktionen der Transaktion wurden durch den externen Sicherheitsmanager gemäß der Benutzersicherheit sowie abhängig davon eingerichtet, ob die Verbindungssicherheit oder ob Execution Diagnostic Facility (EDF) verwendet wurde.

Standardaktion: Die Task wird abnormal beendet.

121 RESUNAVAIL

RESP2-Werte:

121

Eine durch die zu startende Transaktion benötigte Ressource ist in der Zielregion nicht verfügbar. Die Bedingung RESUNAVAIL gilt nur für dynamisch weitergeleitete und nicht auf ein Terminal bezogene EXEC CICS-Anforderungen START.

Die Bedingung RESUNAVAIL wird für den EXEC CICS-Befehl START zurückgegeben, der durch den Spiegel in der Zielregion ausgeführt wird, falls ein Programm für den globalen Benutzerexit XICERES angibt, dass eine erforderliche Ressource in der Zielregion nicht verfügbar ist. An die Anwendung wird die Bedingung nicht zurückgegeben.

Standardaktion: Das verteilte Routing-Programm wird für den Routenauswahlfehler erneut aufgerufen.

53 SYSIDERR

Tritt auf, wenn die Option SYSID einen Namen angibt, bei dem es sich weder um das lokale System noch um ein fernes System handelt (das CICS durch eine CONNECTION-Definition bekanntgegeben wird). Die Bedingung SYSIDERR tritt außerdem auf, wenn die Verbindung zu dem fernen System bekannt, aber nicht verfügbar ist.

Der folgende Fehler wird durch einen RESP2-Wert angegeben:

1

Das Programm für dynamisches Routing hat die Anforderung START zurückgewiesen.

Standardaktion: Die Task wird abnormal beendet.

11 TERMIDERR

Tritt auf, wenn die Terminal-ID in einem Befehl START für CICS nicht definiert ist. Diese Bedingung kann entstehen, falls Sie den Namen einer Verbindung angeben, die keine ISC- oder MRO-Verbindung ist.

Standardaktion: Die Task wird abnormal beendet.

28 TRANSIDERR

Tritt auf, wenn die in einem Befehl START angegebene Transaktions-ID für CICS nicht definiert ist.

Standardaktion: Die Task wird abnormal beendet.

69 USERIDERR

RESP2-Werte:

8

Die angegebene Benutzer-ID ist dem externen Sicherheitsmanager nicht bekannt.

10

Der externe Sicherheitsmanager befindet sich in einem Status, bei dem CICS nicht ermitteln kann, ob eine angegebene Benutzer-ID gültig ist.

19

Die angegebene Benutzer-ID (USERID) wurde entzogen.

Standardaktion: Die Task wird abnormal beendet.

START ATTACH

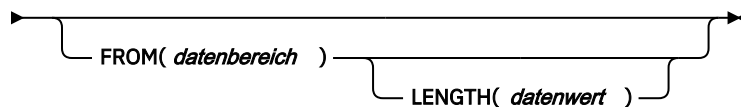
Startet eine Task sofort.

Siehe auch:

- „START“ auf Seite 565
- „START BREXIT“ auf Seite 576
- „START CHANNEL“ auf Seite 579

START ATTACH

►► START ATTACH — TRANSID(*name*) →



Bedingungen: INVREQ, LENGERR, NOTAUTH, TRANSIDERR

Beschreibung

Der Befehl **START ATTACH** startet sofort eine Nicht-Terminal-Task in der lokalen CICS-Region.

Die angehängte Task besitzt den STARTCODE-Wert U und kann nicht abgebrochen werden; EIBREQID ist daher auf Nullen gesetzt.

Der Befehl ATTACH ermöglicht es, dass ein Befehl START, der in einem PLTPI-Programm ausgegeben wird, vor Abschluss der Initialisierung wirksam wird.

Die startende Task kann mit der Option FROM Daten an die gestartete Task übergeben.

Ein Befehl **START ATTACH** gibt nicht den Ursprungsdatensatz (Origin Data Record, ODR) weiter, weshalb Tasks immer an einem neuen Ausgangspunkt gestartet werden.

Tasks mit Daten starten

Falls Daten übergeben werden sollen, werden sie nicht in eine Warteschlange für temporären Speicher eingereiht, sondern es wird lediglich ihre Adresse übergeben.

Die angehängte Task ruft Daten wie üblich ab. Die Task, die den Befehl START ausgegeben hat, muss sicherstellen, dass die Daten zum Zeitpunkt des Abrufs gültig sind, indem sie entweder ihre Ausführung mit der angehängten Task synchronisiert oder die Daten in den gemeinsam genutzten Speicher stellt.

Jeder Befehl **START ATTACH** hat das Starten einer separaten Task zur Folge, optional mit der Übergabe von Daten an die gestartete Task. Das folgende Beispiel zeigt, wie eine angegebene Task gestartet wird und wie Daten an die Task übergeben werden:

```
EXEC CICS START ATTACH  
      TRANSID('TRNL')  
      FROM(DATAFLD)  
      LENGTH(100)  
      :
```

Optionen

FROM(*datenbereich*)

Gibt die Daten an, die an eine gestartete Task übergeben werden sollen.

LENGTH(*datenwert*)

Gibt als Halbwort-Binärdatenwert die Länge der Daten an, die an eine gestartete Task übergeben werden sollen.

TRANSID(*name*)

Gibt die symbolische ID (1 bis 4 Zeichen) der Transaktion an, die durch eine infolge eines Befehls START ATTACH gestarteten Task gestartet werden soll.

Die Option TRANSID wird in erfassten Daten zum vorherigen Hop verwendet. Weitere Informationen finden Sie unter [Merkmale der Daten zum vorherigen Hop](#).

Bedingungen

16 INVREQ

RESP2-Werte:

11

Es wurde versucht, eine Anforderung START ATTACH weiterzuleiten.

12

Eine Anforderung START ATTACH ist fehlgeschlagen.

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

Tritt auf, wenn der Wert für LENGTH nicht größer als Null ist.

Standardaktion: Die Task wird abnormal beendet.

70 NOTAUTH

RESP2-Werte:

7

Eine Ressourcensicherheitsprüfung ist für TRANSID(name) fehlgeschlagen.

Standardaktion: Die Task wird abnormal beendet.

28 TRANSIDERR

Tritt auf, wenn die in einem Befehl START angegebene Transaktions-ID für CICS nicht definiert ist.

RESP2-Werte:

11

Die angegebene Transaktion ist als ferne Transaktion definiert.

Standardaktion: Die Task wird abnormal beendet.

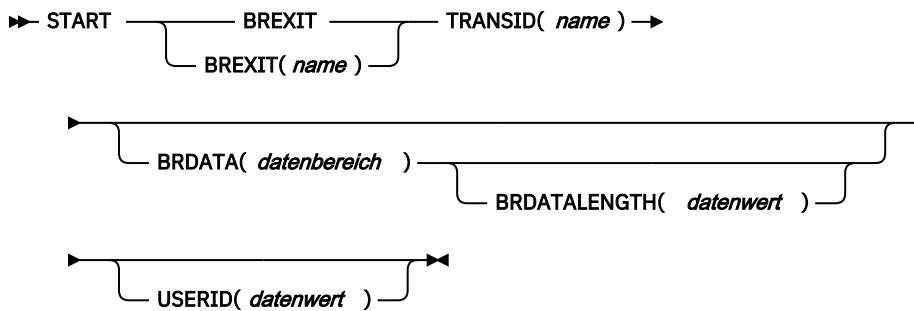
START BREXIT

Startet eine Task in der 3270-Brückenumgebung und ordnet sie dem benannten Brückenexit zu.

Siehe auch:

- [„START“ auf Seite 565](#)
- [„START ATTACH“ auf Seite 575](#)
- [„START CHANNEL“ auf Seite 579](#)

START BREXIT



Bedingungen: INVREQ, LENGERR, NOTAUTH, PGMIDERR, TRANSIDERR, USERIDERR

Beschreibung

Der Befehl **START BREXIT** startet eine Task sofort in der lokalen CICS-Region und initialisiert die angegebene Transaktion (TRANSID) sowie den Brückenexit (BREXIT). In der Umgebung mit 3270-Brücke werden alle 3270-Terminalanforderungen, die von der mit TRANSID angegebenen Transaktion ausgegeben werden, abgefangen und an das durch den Benutzer austauschbare Programm (den Brückenexit) übergeben, das durch die Option BREXIT angegeben ist.

Der Brückenexit (BREXIT) emuliert die 3270-Schnittstelle, indem die Terminalanforderungen an eine Clientanwendung übergeben werden, die innerhalb oder außerhalb von CICS ausgeführt wird.

Weitere Informationen zur 3270-Brücke und ihren Schnittstellen finden Sie unter [Einführung in die 3270-Brücke](#).

Die angehängte Task kann nicht abgebrochen werden; ihr Startcode wird durch den Brückenexit definiert.

Optionen

BREXIT(name)

Gibt den Namen (1 bis 8 Zeichen) des Brückenexits an, der der gestarteten Task zugeordnet werden soll. Falls keine Name angegeben ist, wird der Wert von BREXIT in der TRANSACTION-Ressourcendefinition für die mit TRANSID angegebene Transaktion verwendet.

BRDATA(datenbereich)

Gibt die Daten an, die an den durch BREXIT angegebenen Brückenexit übergeben werden sollen, wenn die Task gestartet wird.

BRDATALENGTH(datenwert)

Gibt als Vollwort-Binärdatenwert die Länge der Daten in der Option BRDATA an, die an den durch BREXIT angegebenen Brückenexit übergeben werden sollen, wenn die Task gestartet wird.

TRANSID(name)

Gibt die symbolische ID (1 bis 4 Zeichen) der Transaktion an, die durch eine infolge eines Befehls START BREXIT gestarteten Task ausgeführt werden soll. Die Transaktion wird in der Umgebung mit 3270-Brücke gestartet und im Zusammenhang mit dem in BREXIT angegebenen Brückenexit ausgeführt.

Die Option TRANSID wird in erfassten Daten zum vorherigen Hop verwendet. Weitere Informationen zur Verwendung der Option TRANSID mit Daten zum vorherigen Hop finden Sie unter [Bezugsdaten](#).

USERID(datenwert)

Gibt die Benutzer-ID an, unter deren Berechtigung die gestartete Transaktion ausgeführt werden soll.

Bedingungen

16 INVREQ

RESP2-Werte:

11

Es wurde versucht, eine Anforderung START BREXIT weiterzuleiten.

12

Eine Anforderung START BREXIT ist fehlgeschlagen.

18

Es wurde ein Wert für die Option USERID angegeben und die CICS-Schnittstelle für den externen Sicherheitsmanager ist nicht initialisiert.

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

Tritt auf, wenn der Wert für BRDATALENGTH nicht größer als Null ist.

Standardaktion: Die Task wird abnormal beendet.

70 NOTAUTH

RESP2-Werte:

7

Eine Ressourcensicherheitsprüfung ist für TRANSID(name) fehlgeschlagen.

9

Eine Sicherheitsprüfung des Ersatzbenutzers ist für USERID(name) fehlgeschlagen. Die Sicherheitszugriffsfunktionen der Transaktion, die den Befehl ausgegeben hat, lassen eine Ausführung des Befehls mit dem Wert, der in der Option USERID angegeben ist, nicht zu.

Standardaktion: Die Task wird abnormal beendet.

27 PGMIDERR

Tritt auf, falls durch die Option BREXIT kein Name angegeben wird und die Transaktionsdefinition für TRANSID keinen Standardnamen für die Option BREXIT bereitstellt.

Standardaktion: Die Task wird abnormal beendet.

28 TRANSIDERR

Tritt auf, falls der in einem Befehl START BREXIT für TRANSID angegebene Wert nicht für CICS definiert wurde.

RESP2-Werte:

11

Die angegebene Transaktion ist als ferne Transaktion definiert.

Standardaktion: Die Task wird abnormal beendet.

69 USERIDERR

RESP2-Werte:

8

Die angegebene Benutzer-ID ist dem externen Sicherheitsmanager nicht bekannt.

10

Der externe Sicherheitsmanager befindet sich in einem Status, bei dem CICS nicht ermitteln kann, ob eine angegebene Benutzer-ID gültig ist.

Standardaktion: Die Task wird abnormal beendet.

Daten an den Brückenexit übergeben

Daten können mit den Optionen BRDATA und BRDATALENGTH an den Brückenexit übergeben werden.

Das folgende Beispiel zeigt, wie eine angegebene Task in der Umgebung mit 3270-Brücke gestartet wird und wie Daten an ihren Brückenexit übergeben werden:

```
EXEC CICS START BREXIT('DFH0CBRE')
      TRANSID('TRNL')
      BRDATA(BRSD)
      BRDATALENGTH(72)
:
```

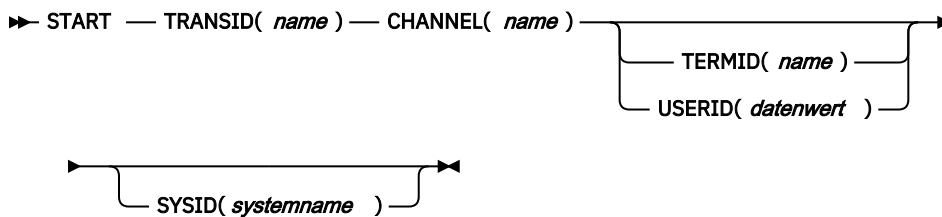
START CHANNEL

Startet eine Task und übergibt sie an einen Kanal.

Siehe auch:

- „START“ auf Seite 565
- „START ATTACH“ auf Seite 575
- „START BREXIT“ auf Seite 576

START CHANNEL



Bedingungen: CHANNELERR, INVREQ, ISCINVREQ, NOTAUTH, RESUNAVAIL, SYSIDERR, TERMIDERR, TRANSIDERR, USERIDERR

Beschreibung

Der Befehl `START CHANNEL` startet eine Task auf einem lokalen oder fernen System und übergibt an sie einen Kanal.

Normalerweise verwendet die startende Task den Kanal, um Daten an die gestartete Task zu übergeben (obwohl der Kanal unter bestimmten Umständen leer sein kann; entsprechende Informationen enthält die Beschreibung der Option CHANNEL). Die startende Task kann außerdem ein Terminal angeben, das von der gestarteten Task als ihre Hauptfunktion verwendet werden soll.

Die gestartete Task kann beispielsweise Folgendes ausführen:

1. Mit dem Befehl **ASSIGN CHANNEL** den Namen eines an sie übergebenen Kanals erkennen
2. Mit den Befehlen **STARTBROWSE CONTAINER CHANNEL** und **GETNEXT CONTAINER** die Container im Kanal durchsuchen
3. Mit den Befehlen **GET CONTAINER CHANNEL** oder **GET64 CONTAINER** auf die Daten in den Containern zugreifen

Die folgenden Bedingungen müssen erfüllt sein, damit die auszuführende Transaktion gestartet werden kann:

- Falls Sie die Option **TERMID** angeben, muss das benannte Terminal vorhanden und verfügbar sein. Wenn das benannte Terminal nicht vorhanden ist, wird der Start verworfen.
- Der Befehl **START CHANNEL** unterstützt **IMS** nicht, Sie können also den Befehl **START CHANNEL** nicht verwenden, um eine Transaktion auf einem fernen **IMS**-System zu starten.

Jeder Befehl START CHANNEL löst den Start einer separaten Task aus.

Ein Befehl **START** mit angegebener Option **TERMID** gibt nicht den Ursprungsdatensatz (Origin Data Record, ODR) weiter, weshalb Tasks immer an einem neuen Ausgangspunkt gestartet werden.

Durch Befehle START gestartete dynamisch weitergeleitete Transaktionen

Einige durch eine Untergruppe von Befehlen START gestartete Transaktionen können dynamisch an eine ferne Region weitergeleitet werden. Allgemeine Angaben über das dynamische Transaktionsrouting und spezielle Informationen dazu, welche durch Befehle START gestarteten Transaktionen für das dynamische Routing infrage kommen, finden Sie unter [Mit START-Befehlen aufgerufene Transaktionen weiterleiten](#).

Fehler bei Befehlen START ohne Ausnahmebedingungen

Unter den folgenden Umständen wird ein Befehl START zwar ohne Fehler ausgeführt, die gestartete Task findet jedoch zu keinem Zeitpunkt statt:

- Die Transaktion oder ihr Arbeitsgang ist zu dem Zeitpunkt nicht verfügbar, an dem CICS versucht, die Task zu erstellen.
- Der Befehl START gibt ein Terminal an, das zu dem Zeitpunkt, an dem CICS versucht, die Task zu erstellen, nicht definiert ist (und nicht durch die Exits XICTENF oder XALTENF gefunden werden kann).
- Falls das angeforderte Terminal zum Zeitpunkt des Befehls START nicht vorhanden ist, erhalten Sie eine Bedingung TERMIDERR. Wird das Terminal jedoch später nicht verfügbar, was beim Abmelden des Benutzers passiert, wird Ihre Anforderung verworfen und es tritt keine Bedingung TERMIDERR auf.

Diese Risiken entstehen durch die Verzögerung zwischen der Ausführung des Befehls START und dem Zeitpunkt der Taskerstellung. Selbst wenn der Befehl START bei einer Anforderung START CHANNEL sofort ausgeführt wird, ist es möglich, dass CICS die Erstellung der Task verzögert, weil entweder das angeforderte Terminal nicht verfügbar ist oder andere Systembedingungen vorliegen.

Mit Befehlen INQUIRE können Sie sicherstellen, dass die Transaktion und das Programm zum Zeitpunkt des Befehls START aktiviert sind. Es kann allerdings sein, dass die Verfügbarkeit der Transaktion oder des Programms vor der Taskerstellung endet.

Optionen

CHANNEL(name)

Gibt den 1 bis 16 Zeichen langen Namen eines Kanals an, der für die gestartete Task verfügbar gemacht werden soll. Zulässige Zeichen sind A-Z a-z 0-9 \$ @ # ./- _ % & ? ! : | " = ~ , ; < > . Führende und eingebettete Leerzeichen sind nicht zulässig. Falls der angegebene Name weniger als 16 Zeichen umfasst, wird er mit nachgestellten Leerzeichen auf 16 Zeichen aufgefüllt. Falls der Kanal nicht vorhanden ist, wird er erstellt. Dieser neue Kanal bleibt gültig, bis sich die Verbindungsebene ändert. Informationen zum Geltungsbereich von Kanälen finden Sie unter [Geltungsbereich eines Kanals](#).

Kanalnamen werden immer in EBCDIC angegeben. Die oben angegebene Gruppe der zulässigen Zeichen für Kanalnamen enthält einige Zeichen, die nicht in allen EBCDIC-Codepages dieselbe Darstellung besitzen. Falls Kanäle zwischen Regionen verlagert werden sollen, empfiehlt es sich daher, bei ihrer Benennung ausschließlich die Zeichen A-Z a-z 0-9 & : = , ; < > . - und _ zu verwenden.

Sie können den Kanalnamen DFHTRANSACTION angeben, um einen Transaktionskanal zu verwenden. Ein Transaktionskanal verliert nicht seine Gültigkeit, wenn sich die Verbindungsebene ändert; der Zugriff auf diesen Kanal ist in der Transaktion immer möglich. Weitere Informationen finden Sie unter [Kanäle und Container](#).

Das Programm, das den Befehl START ausgibt, kann Folgendes vornehmen:

- Es kann mit einem oder mehreren Befehlen **PUT CONTAINER CHANNEL** bzw. **PUT64 CONTAINER** den Kanal erstellen, bevor es den Befehl **START** ausgibt.
- Es kann seinen aktuellen Kanal durch den Namen angeben.
- Es kann einen Kanal benennen, der zurzeit nicht vorhanden ist. In diesem Fall wird ein neuer leerer Kanal erstellt.

Die gestartete Task erhält eine *Kopie* der Container des Kanals (und der in ihnen enthaltenen Daten). Die Kopie wird bei der Ausgabe des Befehls START erstellt.

Anmerkung: Alle EXEC CICS-Anforderungen START, die einen Kanal angeben, können nicht durch die Angabe von **INTERVAL**, **AT**, **FOR** oder **UNTIL** verzögert werden, da dies nicht unterstützt wird.

SYSID(systemname)

Gibt den Namen des Systems an, an das die Anforderung gerichtet ist.

TERMID(name)

Gibt die symbolische ID (1 bis 4 alphanumerische Zeichen) der Hauptfunktion an, die einer Transaktion zugeordnet ist, die infolge eines Befehls START gestartet werden soll. Diese Hauptfunktion kann entweder ein Terminal (im Normalfall) oder eine APPC-Sitzung sein. Bei Angabe einer APPC-Sitzung wird anstelle einer Terminal-ID der Name der Verbindung (oder des Modussets) verwendet. Diese Option ist erforderlich, wenn die zu startende Transaktion mit einem Terminal kommunizieren muss; andernfalls sollte sie weggelassen werden.

Sie müssen die Terminal-ID entweder als lokales oder als fernes Terminal in dem System definieren, in dem der Befehl START ausgeführt werden soll.

Die Option TERMID wird in erfassten Daten zum vorherigen Hop verwendet. Weitere Informationen finden Sie unter [Merkmale der Daten zum vorherigen Hop](#).

Verwenden Sie die Option TERMID nicht, wenn Sie die Identitätsweitergabe einsetzen.

TRANSID(name)

Gibt die symbolische ID (1 bis 4 Zeichen) der Transaktion an, die durch eine infolge eines Befehls START gestarteten Task ausgeführt werden soll.

Falls Sie die Option SYSID angeben und ein fernes System benennen, wird davon ausgegangen, dass sich die Transaktion unabhängig davon, ob sie als fern definiert ist, auf diesem System befindet. Andernfalls wird mithilfe der Transaktionsdefinition ermittelt, ob sich die Transaktion auf einem lokalen oder auf einem fernen System befindet.

Die Option TRANSID wird in erfassten Daten zum vorherigen Hop verwendet. Weitere Informationen finden Sie unter [Merkmale der Daten zum vorherigen Hop](#).

USERID(datenwert)

Gibt die Benutzer-ID an, unter deren Berechtigung die gestartete Transaktion ausgeführt werden soll, wenn die gestartete Transaktion keinem Terminal zugeordnet ist (die Option TERMID also nicht angegeben wurde). Diese Benutzer-ID wird als *userid1* bezeichnet.

Wenn Sie weder die Option TERMID noch die Option USERID angeben, verwendet CICS stattdessen die Benutzer-ID, unter der die Transaktion ausgeführt wird, die den Befehl START ausgibt. Diese Benutzer-ID wird als *userid2* bezeichnet.

Durch die Verwendung von *userid1* oder *userid2* stellt CICS sicher, dass eine gestartete Transaktion immer unter einer gültigen Benutzer-ID ausgeführt wird, die für alle Ressourcen berechtigt sein muss, die von der gestarteten Transaktion referenziert werden.

CICS führt für *userid2* eine Ersatzsicherheitsprüfung durch, um zu verifizieren, dass dieser Benutzer für *userid1* berechtigt ist. Falls *userid2* nicht berechtigt ist, gibt CICS eine Bedingung NOTAUTH zurück. Die Ersatzprüfung wird hier nicht durchgeführt, falls die Option USERID nicht angegeben ist.

Verwenden Sie die Option USERID nicht, wenn Sie die Identitätsweitergabe einsetzen.

Bedingungen

122 CHANNELERR

RESP2-Werte:

1

Der in der Option CHANNEL angegebene Kanal enthält ein falsches Zeichen oder eine falsche Zeichenkombination.

16 INVREQ

RESP2-Werte:

9

Die im Befehl angegebenen Optionen sind nicht kompatibel.

17

Die gestartete Transaktion ist nicht für einen Systemabschluss geeignet und für die CICS-Region wird gerade ein Systemabschluss durchgeführt.

18

Es wurde ein Wert für USERID angegeben und die CICS-Schnittstelle für den externen Sicherheitsmanager ist nicht initialisiert.

200

Der Befehl START wurde durch ein Serverprogramm für Distributed Program Link (DPL) ausgegeben und verwendete die Option TERMID, deren Wert mit der ID der systemübergreifenden Sitzung übereinstimmt. Wenn der Wert von TERMID mit dem Wert von EIBTRMID identisch ist (es sich bei der Hauptfunktion der ausgebenden Task also nicht um ein Terminal, sondern um eine Sitzung handelt), gehört der Befehl START TERMID nicht zu der für DPL-Serverprogramme verfügbaren Teilmenge der zulässigen Befehle.

Die Bedingung INVREQ (RESP2 nicht festgelegt) tritt ebenfalls auf, falls der Befehl START nicht für die Verarbeitung durch CICS gültig ist.

Standardaktion: Die Task wird abnormal beendet.

54 ISCINVREQ

Tritt auf, wenn das ferne System einen Fehler angegeben hat, der keiner bekannten Bedingung entspricht.

Standardaktion: Die Task wird abnormal beendet.

70 NOTAUTH

RESP2-Werte:

7

Eine Ressourcensicherheitsprüfung ist für TRANSID(name) fehlgeschlagen.

9

Eine Sicherheitsprüfung des Ersatzbenutzers ist für USERID(name) fehlgeschlagen.

Die Sicherheitszugriffsfunktionen der Transaktion, die den Befehl ausgegeben hat, lassen eine Ausführung des Befehls mit dem Wert, der in der Option USERID angegeben ist, nicht zu. Die Sicherheitszugriffsfunktionen der Transaktion wurden durch den externen Sicherheitsmanager gemäß der Benutzersicherheit sowie abhängig davon eingerichtet, ob die Verbindungssicherheit oder ob Execution Diagnostic Facility (EDF) verwendet wurde.

Standardaktion: Die Task wird abnormal beendet.

121 RESUNAVAIL

RESP2-Werte:

121

Eine durch die zu startende Transaktion benötigte Ressource ist in der Zielregion nicht verfügbar. Die Bedingung RESUNAVAIL gilt nur für *dynamisch weitergeleitete und nicht auf ein Terminal bezogene* EXEC CICS-Anforderungen START.

Die Bedingung RESUNAVAIL wird für den EXEC CICS-Befehl START zurückgegeben, der *durch den Spiegel in der Zielregion ausgeführt* wird, falls ein Programm für den globalen Benutzerexit XICERES angibt, dass eine erforderliche Ressource in der Zielregion nicht verfügbar ist. An die Anwendung wird die Bedingung nicht zurückgegeben.

Standardaktion: Das verteilte Routing-Programm wird für den Routenauswahlfehler erneut aufgerufen.

53 SYSIDERR

Tritt in allen folgenden Fällen auf:

- Die Option SYSID gibt einen Namen an, bei dem es sich weder um das lokale System noch um ein fernes System handelt (das CICS durch eine CONNECTION-Definition bekanntgegeben wird).
- Der Link zu dem fernen System ist bekannt, aber nicht verfügbar.

In allen obigen Fällen wird die Gattung des Fehlers durch das zweite Byte im Feld EIBRCODE angegeben.

Die folgenden Fehler werden durch RESP2-Werte angegeben:

1

Das Programm für dynamisches Routing hat die Anforderung START zurückgewiesen.

2

Die Option CHANNEL wurde verwendet und die Anforderung START wurde einem fernen System zugestellt bzw. an ein fernes System weitergeleitet, von dem sie nicht unterstützt wird. (Nur bei MRO-Verbindungen.)

Standardaktion: Die Task wird abnormal beendet.

20

Die Option CHANNEL wurde angegeben und die Anforderung START soll über eine LUTYPE61-Verbindung zugestellt werden. Die Zustellung von Anforderungen START CHANNEL über LUTYPE61-Verbindungen ist nicht möglich.

11 TERMIDERR

Die Terminal-ID in einem Befehl START ist für CICS nicht definiert.

Standardaktion: Die Task wird abnormal beendet.

28 TRANSIDERR

Die in einem Befehl START angegebene Transaktions-ID ist nicht für CICS definiert.

Standardaktion: Die Task wird abnormal beendet.

69 USERIDERR

RESP2-Werte:

8

Die angegebene Benutzer-ID ist dem externen Sicherheitsmanager nicht bekannt.

10

Der externe Sicherheitsmanager befindet sich in einem Status, bei dem CICS nicht ermitteln kann, ob eine angegebene Benutzer-ID gültig ist.

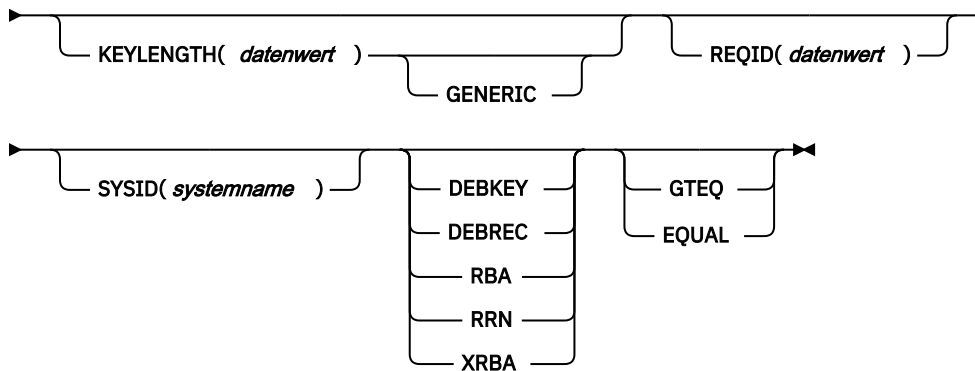
Standardaktion: Die Task wird abnormal beendet.

STARTBR

Startet das Durchsuchen einer Datei.

STARTBR

➔ STARTBR — FILE(*dateiname*) — RIDFLD(*datenbereich*) —➔



Bedingungen: DISABLED, FILENOTFOUND, ILLOGIC, INVREQ, IOERR, ISCINVREQ, LOADING, NOTAUTH, NOTFND, NOTOPEN, SYSIDERR

Dieser Befehl ist threadsicher, wenn für die Datei, auf die er sich bezieht, Folgendes zutrifft:

- Sie ist als fern definiert und der Befehl wird über eine IPIC-Verbindung durch Funktionsverlagerung an eine ferne CICS-Region übertragen.
- Sie ist als lokale VSAM-Datei, als VSAM-RLS-Datei oder als Coupling-Facility-Datentabelle definiert.

Dieser Befehl ist nicht threadsicher, wenn für die Datei, auf die er sich bezieht, Folgendes zutrifft:

- Sie ist als fern definiert und der Befehl wird über eine Nicht-IPIC-Verbindung durch Funktionsverlagerung übertragen.
- Sie ist als gemeinsam genutzte Datentabelle oder als BDAM-Datei definiert.

Beschreibung

Der Befehl STARTBR gibt den Datensatz in einer Datei oder Datentabelle auf einem lokalen oder fernen System an, bei dem der Suchvorgang beginnen soll. Datensätze werden erst dann gelesen, wenn ein Befehl READNEXT (bzw. bei VSAM und Tabellen ein Befehl READPREV) ausgeführt wird.

Die folgenden Suchvorgänge sind möglich. Bei einem direkten Suchvorgang wird die Basisdatei mithilfe des Primärschlüssels durchsucht.

- Direkter Suchvorgang in einer schlüsseladressierten Datei (KSDS oder Datentabelle) nach Satzschlüssel
- Direkter Suchvorgang in einer Datei in Zugangsfolge (Entry-Sequenced Data Set, ESDS) nach relativer Byteadresse (RBA)
- Direkter Suchvorgang in einer Datei für relative Datensätze (Relative Record Data Set, RRDS) nach relativer Satznummer (RRN)
- Direkter Suchvorgang in einer schlüsseladressierten Datei (KSDS) unter Verwendung eines Alternativindexpfades
- Suchvorgang in einer Datei in Zugangsfolge unter Verwendung eines Alternativindexpfades (In diesem Fall wird eine ESDS auf dieselbe Weise nach dem Schlüssel durchsucht wie eine KSDS. Einige der bei einem direkten ESDS-Suchvorgang nicht gültigen Optionen sind bei Suchvorgängen mit Alternativindex gültig.)
- Suchvorgang in einer schlüsseladressierten Datei nach relativer Byteadresse

Die im Befehl STARTBR angegebenen Optionen definieren die Merkmale, die für die gesamte nachfolgende Suchoperation gelten. Insbesondere werden gegebenenfalls angegebene Optionen GENERIC oder GTEQ nicht nur zum Ermitteln des Ausgangspunktes für den Suchvorgang, sondern auch immer dann verwendet, wenn sich der Wert von RIDFLD ändert, bevor ein Befehl READNEXT ausgegeben wird.

Falls Sie die Option RBA angeben, gilt diese für jeden Befehl READNEXT oder READPREV im Suchvorgang und führt dazu, dass CICS für jeden abgerufenen Datensatz die relative Byteadresse zurückgibt.

Außer durch einen Befehl RESETBR kann keine dieser Optionen während eines Suchvorgangs geändert werden.

Falls eine Anforderung STARTBR den exakten Schlüssel angibt, bei dem der Suchvorgang gestartet werden soll (sie also den vollständigen Schlüssel und das Schlüsselwort EQUAL angibt), wird beim folgenden Befehl READNEXT (bzw. READPREV) möglicherweise nicht derselbe Datensatz zurückgegeben, der durch den Befehl STARTBR für eine im NSR- oder RLS-Modus von VSAM geöffnete Datei angegeben wird. Dies kann auftreten, weil der ursprünglich im Befehl STARTBR angegebene Datensatz zwischen der Fertigstellung des Befehls STARTBR und der Ausgabe eines Befehls READNEXT oder READPREV durch eine andere Transaktion gelöscht werden kann. Im LSR-Modus von VSAM kann der ursprüngliche Datensatz zwischen Befehlen STARTBR und READNEXT nicht gelöscht werden.

Ein Suchvorgang kann durch die Befehle ENDBR, SYNCPOINT oder SYNCPOINT ROLLBACK beendet werden. Auch ein impliziter Synchronisationspunkt am Ende der Task beendet den Suchvorgang.

Optionen

DEBKEY

(Bei BDAM-Blockung) Gibt an, dass die Entblockung nach Schlüssel erfolgen soll. Falls weder DEBREC noch DEBKEY angegeben ist, findet keine Entblockung statt.

DEBREC

(Bei BDAM-Blockung) Gibt an, dass die Entblockung nach relativem Datensatz (bezogen auf Null) erfolgen soll. Falls weder DEBREC noch DEBKEY angegeben ist, findet keine Entblockung statt.

EQUAL

(VSAM und Datentabelle) Gibt an, dass die Suche nur durch einen Datensatz erfüllt wird, dessen (vollständiger oder generischer) Schlüssel ebenfalls in der Option RIDFLD angegeben ist.

Diese Option ist das Standardfeld für einen direkten ESDS-Suchvorgang.

FILE(dateiname)

Gibt den Namen der Datei an, auf die zugegriffen werden soll.

Bei Angabe eines Wertes für SYSID wird davon ausgegangen, dass sich die Datei, auf die sich dieser Dateiname bezieht, auf einem fernen System befindet, wobei es ohne Belang ist, ob der Name für CICS definiert ist. Ansonsten wird die Ressourcendefinition verwendet, um zu ermitteln, ob sich die Datei (Dataset) auf einem lokalen oder fernen System befindet.

GENERIC

(Schlüsseladressierte VSAM-Datei (KSDS), Pfad oder Datentabelle) Gibt an, dass es sich bei dem Suchkriterium um einen generischen Schlüssel handelt, dessen Länge in der Option KEYLENGTH angegeben ist. Die Suche nach einem Datensatz wird erfüllt, wenn ein Datensatz gefunden wird, der dieselben Anfangszeichen (generischer Schlüssel) wie hier angegeben hat.

GTEQ

(VSAM oder Datentabelle) Gibt an, dass bei einer erfolglosen Suche nach einem Datensatz, der denselben (vollständigen oder generischen) Schlüssel wie in der Option RIDFLD angegeben aufweist, der erste Datensatz die Suche erfüllt, der einen größeren Schlüssel besitzt.

Diese Option ist die Standardeinstellung bei einem direkten Suchvorgang für eine KSDS- oder RRDS-Datei. Bei einem direkten Suchvorgang für eine ESDS-Datei ist sie nicht gültig; beim Suchvorgang für eine ESDS-Datei unter Verwendung eines Pfades ist sie jedoch gültig.

KEYLENGTH(datenwert)

Gibt die Länge (Halbwort-Binärzahl) des Schlüssels an, der in der Option RIDFLD angegeben ist. Ausgenommen hiervon ist die Angabe der relativen Byteadresse (RBA) oder der relativen Satznummer (RRN); in diesem Fall ist die Option KEYLENGTH nicht gültig.

Diese Option muss bei Angabe von GENERIC angegeben werden; sie kann immer dann angegeben werden, wenn ein Schlüssel angegeben ist. Falls die angegebene Länge von der für die Datei definier-

ten Länge abweicht und es sich nicht um eine generische Operation handelt, tritt die Bedingung INVREQ auf.

Die Bedingung INVREQ tritt ebenfalls auf, falls ein Befehl STARTBR die Option GENERIC angibt und der Wert von KEYLENGTH nicht kleiner als der Wert ist, der in der VSAM-Definition angegeben ist.

Falls die Option KEYLENGTH(0) verwendet wird, um den ersten Datensatz in der Datei zu positionieren, muss auch die Option GTEQ angegeben werden. Falls EQUAL entweder explizit oder standardmäßig mit KEYLENGTH(0) angegeben wird, sind die Ergebnisse des Befehls STARTBR unvorhersehbar.

Für ferne Dateien kann der Wert von KEYLENGTH in der FILE-Definition angegeben werden. Falls die Option KEYLENGTH dort nicht definiert ist, nicht im Anwendungsprogramm angegeben ist und der Schlüssel länger als 4 Zeichen ist, lautet der Standardwert 4.

RBA

(Nur schlüsseladressierte VSAM-Dateien oder ESDS-basierte Dateien bzw. CICS-verwaltete Datentabellen, jedoch keine Pfade) Gibt an, dass das Feld für die Satzkennung in der Option RIDFLD eine relative Byteadresse enthält. Verwenden Sie diese Option nur dann, wenn Sie eine ESDS- oder KSDS-Basis durchsuchen und die Datensätze nicht mit Schlüsseln, sondern mit relativen Byteadressen angeben.

Die Option RBA kann für Folgendes nicht verwendet werden:

- Benutzerverwaltete Datentabellen
- Coupling-Facility-Datentabellen
- Jede KSDS-Datei, die im RLS-Zugriffsmodus geöffnet ist
- KSDS-Dateien, die erweiterte Adressierung verwenden

Von einer Verwendung der Option RBA wird darüber hinaus bei ESDS-Dateien mit einer Größe von mehr als 4 GB abgeraten. (Verwenden Sie stattdessen XRBA.)

REQID(datenwert)

Gibt in Form eines Halbwort-Binärwertes eine eindeutige Anforderungskennung für einen Suchvorgang an, mit dessen Hilfe Mehrfachsuchvorgänge für dieselben oder andere Dateien gesteuert werden. Falls diese Option nicht angegeben ist, wird vom Standardwert Null ausgegangen.

RIDFLD(datenbereich)

Gibt das Feld für die Satzkennung an. Der Inhalt kann ein Schlüssel, eine relative Byteadresse, eine relative Satznummer (für VSAM-Dateien) oder ein Blockverweis, ein physischer Schlüssel und ein Entblockungsargument (für BDAM-Dateien) sein. Bei einer relativen Byteadresse oder einer relativen Satznummer muss dieses Feld das Format einer Vollwort-Binärzahl besitzen. Für eine relative Byteadresse kann der Wert von RIDFLD größer-gleich 0 sein. Für eine relative Satznummer kann der Wert von RIDFLD größer-gleich 1 sein.

Weitere Informationen zum Definieren des Feldes für die Satzkennung finden Sie unter [BDAM-Datensätze angeben](#) und [VSAM-Datensätze angeben](#).

Bei VSAM gibt eine vollständige Datensatz-ID aus Werten X'FF' an, dass der Suchvorgang zur Vorbereitung einer Rückwärtssuche mit Befehlen READPREV am Ende der Datei positioniert werden soll.

RRN

(Schlüsseladressierte VSAM-Dateien) Gibt an, dass das in der Option RIDFLD angegebene Feld für die Satzkennung eine relative Satznummer enthält. Diese Option sollte nur im Zusammenhang mit Dateien verwendet werden, die relative Datensatzdateien referenzieren.

SYSID(systemname)

Gibt den Namen des Systems an, an das die Anforderung gerichtet ist.

Falls Sie die Option SYSID angeben und beide Optionen RBA und RRN weglassen, müssen Sie ebenfalls die Option KEYLENGTH angeben.

XRBA

Gibt an, dass das in der Option RIDFLD angegebene Feld für die Satzkennung eine erweiterte relative Byteadresse enthält. Diese Option sollte bei Suchvorgängen für Datensätze in einer ESDS-Datei mit erweiterter Adressierung verwendet werden.

Falls Sie die Option XRBA in einem Befehl STARTBR angeben, müssen alle anderen Befehle innerhalb desselben Suchvorgangs ebenfalls XRBA angeben.

Der Zugriff auf schlüsseladressierte Dateien mit XRBA ist nicht möglich.

Bedingungen

84 DISABLED

RESP2-Werte:

50

Eine Datei ist inaktiviert. Dies kann die folgenden Gründe haben:

- Sie wurde anfänglich als inaktiviert definiert und wurde seither nicht aktiviert.
- Die Datei wurde durch einen Befehl SET FILE oder CEMT SET FILE inaktiviert.

Standardaktion: Die Task wird abnormal beendet.

12 FILENOTFOUND

RESP2-Werte:

1

Ein in der Option FILE referenzierter Dateiname ist nicht für CICS definiert und die Option SYSID wurde nicht angegeben.

Standardaktion: Die Task wird abnormal beendet.

21 ILLOGIC

RESP2-Werte (VSAM):

110

Es ist ein VSAM-Fehler aufgetreten, der zu keiner der anderen CICS-Antwortkategorien gehört.

Weitere Informationen sind bei EIBRCODE im EXEC-Schnittstellenblock verfügbar; Details finden Sie unter [EIB-Felder](#).

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

RESP2-Werte:

20

Gemäß der Ressourcendefinition sind Suchoperationen nicht zulässig.

25

Die Optionen KEYLENGTH und GENERIC sind angegeben und die in der Option KEYLENGTH definierte Länge für die Datei, die diese Dateiangabe referenziert, ist größer-gleich der Länge eines vollständigen Schlüssels.

26

Die Option KEYLENGTH wurde angegeben (jedoch nicht die Option GENERIC) und die angegebene Schlüssellänge ist nicht mit der Länge identisch, die für die Datei definiert ist, auf die sich diese Dateiangabe bezieht.

33

Es wurde versucht, einen Suchvorgang mit einem Wert für REQID zu starten, der bereits durch einen anderen Suchvorgang verwendet wird.

42

Die Optionen KEYLENGTH und GENERIC sind angegeben und die in der Option KEYLENGTH angegebene Länge ist kleiner als Null.

44

Die angegebene Datei ist eine benutzerverwaltete oder Coupling-Facility-Datentabelle und der Befehl hat nicht das Format eines Befehls STARTBR für eine solche Datentabelle (beispielsweise ist die relative Byteadresse (RBA) angegeben).

51

Ein Befehl STARTBR für eine schlüsseladressierte Datei, auf die im RLS-Modus zugegriffen wird, gibt das Schlüsselwort RBA an. Der RLS-Modus unterstützt für Datensätze in schlüsseladressierten Dateien keinen Zugriff über die relative Byteadresse (RBA).

59

XRBA wurde angegeben, jedoch ist die Datei (Dataset) keine erweiterte ESDS-Datei.

Standardaktion: Die Task wird abnormal beendet.

17 IOERR

RESP2-Werte:

120

Während der Dateisteuerungsoperation ist ein E/A-Fehler aufgetreten. Ein E/A-Fehler ist ein nicht erwartetes Ereignis, das nicht durch eine CICS-Bedingung abgedeckt ist.

Bei VSAM-Dateien gibt die Bedingung IOERR für gewöhnlich einen Hardwarefehler an. Weitere Informationen sind im EXEC-Schnittstellenblock verfügbar; Details finden Sie unter [EIB-Felder](#).

Bei einer Coupling-Facility-Datentabelle gibt eine Bedingung IOERR an, dass eine falsche Antwort von einem Coupling-Facility-Zugriff zurückgegeben wurde.

Standardaktion: Die Task wird abnormal beendet.

54 ISCINVREQ

RESP2-Werte:

70

Das ferne System gibt einen Fehler an, der keiner bekannten Bedingung entspricht.

Standardaktion: Die Task wird abnormal beendet.

94 LOADING

RESP2-Werte:

104

Die Anforderung kann nicht erfüllt werden, weil sie für eine Datentabelle ausgegeben wurde, die noch geladen wird. Die Bedingung kann aus einem der folgenden Gründe auftreten:

- Der Befehl STARTBR gibt einen Datensatz an, der noch nicht in eine Coupling-Facility-Datentabelle geladen wurde. Datensätze können während des Ladevorgangs einer Coupling-Facility-Datentabelle nur dann durchsucht werden, falls der angeforderte Schlüssel im Bereich derjenigen Datensätze zu finden ist, die bereits geladen worden sind.

Die Antwort LOADING kann auch für eine Coupling-Facility-Datentabelle zurückgegeben werden, deren Ladevorgang fehlgeschlagen ist. Weitere Informationen zu den Folgen, die sich beim Fehlschlagen des Ladevorgangs für eine Coupling-Facility-Datentabelle ergeben, können Sie der Beschreibung für den globalen Benutzerexit XDTLC unter [Exits zur Datentabellenverwaltung](#) (XDTRD, XDTAD und XDTLC) entnehmen.

- Der Befehl READ hat die Optionen GENERIC oder GTEQ für eine benutzerverwaltete Datentabelle angegeben. Während eine benutzerdefinierte Datentabelle geladen wird, können Sie Anforderungen zum Starten von Suchvorgängen ausschließlich mit exakten Schlüsseln ausgeben.

Falls Ihr Anwendungsprogramm die Bedingung LOADING ständig oder zu häufig feststellt, vergewissern Sie sich, dass dies nicht durch kollidierende FILE-Definitionen verursacht wird, die dieselbe Datei referenzieren.

Standardaktion: Die Task wird abnormal beendet.

70 NOTAUTH

RESP2-Werte:

101

Eine Ressourcensicherheitsprüfung ist für FILE(dateiname) fehlgeschlagen.

Standardaktion: Die Task wird abnormal beendet.

13 NOTFND

RESP2-Werte:

80

Der Versuch, die Position für einen Datensatz anhand des angegebenen Sucharguments festzulegen, war erfolglos.

81

Die Option XRBA war angegeben und der Wert von RIDFLD war größer als 4 GB, bei der Datei handelt es sich jedoch nicht um eine ESDS-Datei mit erweiterter Adressierung.

Die Bedingung NOTFND kann außerdem auftreten, wenn ein generischer Befehl STARTBR mit der Option KEYLENGTH(0) auch die Option EQUAL angibt.

Standardaktion: Die Task wird abnormal beendet.

19 NOTOPEN

RESP2-Werte:

60

Die Bedingung NOTOPEN (RESP2 60) wird aus einem der folgenden Gründe zurückgegeben:

- Die angeforderte Datei ist geschlossen (CLOSED) und nicht aktiviert (UNENABLED). Der Status CLOSED, UNENABLED wird erreicht, wenn eine CLOSE-Anforderung für eine geöffnete aktivierte (OPEN ENABLED) Datei empfangen wurde und die Datei nicht mehr im Gebrauch ist. Sie können CLOSED, UNENABLED auch als Anfangsstatus festlegen, indem Sie in der FILE-Ressourcendefinition STATUS(UNENABLED) und OPENTIME(FIRSTREF) angeben. (Bei BDAM-Dateien verwenden Sie den Parameter FILSTAT des Makros DFHFCT (TYPE=FILE)).
- Die angeforderte Datei besitzt den Status OPEN (geöffnet) und wird durch andere Transaktionen verwendet, aber für die Datei wurde eine Anforderung CLOSE empfangen.
- Ein Befehl STARTBR wurde für eine Datei ausgegeben, die sich aufgrund eines Befehls SET DSNAME QUIESCED oder IMMQUIESCED im Quiescemodus befindet oder gerade in diesen Modus versetzt wird.
- Die angeforderte Datei hat den Status CLOSED (geschlossen) und ENABLED (aktiviert), weshalb CICS versucht hat, die Datei im Rahmen der Anforderungsausführung zu öffnen. Das Öffnen der Datei ist aus einem nicht näher bezeichneten Grund fehlgeschlagen. Sie sollten in der Konsole nach Nachrichten suchen, die erläutern, warum die Datei nicht erfolgreich geöffnet werden konnte.

Diese Bedingung tritt nicht auf, wenn die Anforderung für eine geschlossene, nicht aktivierte Datei (CLOSED, DISABLED) abgesetzt wird. In diesem Fall tritt die Bedingung DISABLED auf.

Standardaktion: Die Task wird abnormal beendet.

53 SYSIDERR

RESP2-Werte:

130

Die Option SYSID gibt einen Namen an, bei dem es sich weder um das lokale System noch um ein fernes System handelt, das durch eine CONNECTION- oder IPCONN-Definition definiert ist. Die Bedingung SYSIDERR tritt außerdem auf, wenn die Verbindung zu dem fernen System bekannt, aber nicht verfügbar ist. Bei einer IPCONN-Definition tritt die Bedingung SYSIDERR auf, falls die Verbindung bekannt ist, jedoch entweder das lokale oder das ferne System keine Dateisteuerungsbeefehle unterstützen, die mit IP-Interkonnektivität per Funktionsverlagerung übertragen werden.

131

Für eine Coupling-Facility-Datentabelle ist die Verbindung zum Server für Coupling-Facility-Datentabellen fehlgeschlagen. Dies könnte dadurch verursacht worden sein, dass der Server selbst fehl-

132

Standardaktion: Die Task wird abnormal beendet.

Startet das Durchsuchen (Browsing) der untergeordneten Aktivitäten einer BTS-Aktivität oder eines BTS-Prozesses.

In der Regel wurde die Aktivitäts-ID, die in dieser Option angegeben wird, in einem vorherigen Befehl GETNEXT ACTIVITY (oder, im Fall einer Stammaktivität, in einem Befehl GETNEXT PROCESS) zurückgegeben. Mit der Option ACTIVITYID können Sie ein Durchsuchen der untergeordneten Aktivitäten eine Ebene weiter unten in der Aktivitätsbaumstruktur starten.

Wenn Sie weder diese Option noch die Option PROCESS angeben, werden die untergeordneten Aktivitäten der aktuellen Aktivität durchsucht.

BROWSETOKEN(datenbereich)

Gibt den Datenbereich (Vollwort-Binärwert) an, in den CICS das Browse-Token einfügt.

PROCESS(datenwert)

Gibt den Namen (1 - 36 Zeichen) des Prozesses an, dessen untergeordnete Aktivitäten durchsucht werden sollen.

PROCESSTYPE(datenwert)

Gibt den Prozesstyp (1 - 8 Zeichen) des Prozesses an, der in der Option PROCESS angegeben wird.

Bedingungen

109 ACTIVITYERR

RESP2-Werte:

1

Die in der Option ACTIVITYID angegebene Aktivität konnte nicht gefunden werden.

2

Da weder die Option ACTIVITYID noch die Optionen für den Prozess angegeben wurden, wurde ein Durchsuchen der untergeordneten Aktivitäten der aktuellen Aktivität impliziert, jedoch ist der Anforderung keine aktuelle Aktivität zugeordnet.

19

Die Anforderung hat das Zeitlimit überschritten. Es kann sein, dass die Beendigung einer anderen Task, die diesen Aktivitätsdatensatz verwendet, verhindert wurde.

29

Die Repository-Datei ist nicht verfügbar.

30

Es ist ein Ein-/Ausgabefehler für die Repository-Datei aufgetreten.

70 NOTAUTH

RESP2-Werte:

101

Der Benutzer, der der absetzenden Task zugeordnet ist, ist nicht berechtigt, auf die Datei zuzugreifen, deren Datei (Dataset) die zu durchsuchenden Datensätze enthält.

108 PROCESSERR

RESP2-Werte:

3

Der in der Option PROCESS angegebene Prozess konnte nicht gefunden werden.

4

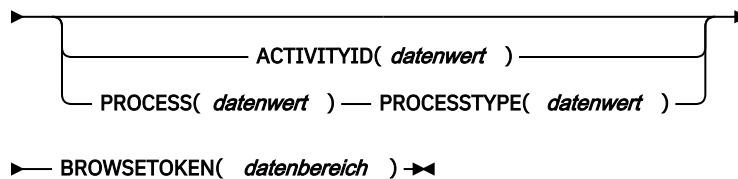
Der in der Option PROCESSTYPE angegebene Prozesstyp konnte nicht gefunden werden.

STARTBROWSE CONTAINER (BTS)

Startet ein Durchsuchen (Browsing) der Container, die einer BTS-Aktivität oder einem BTS-Prozess zugeordnet sind.

STARTBROWSE CONTAINER

➡ STARTBROWSE — CONTAINER →



Bedingungen: ACTIVITYERR, IOERR, NOTAUTH, PROCESSERR

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl STARTBROWSE CONTAINER initialisiert ein Browse-Token, mit dem der Name jedes Datencontainers identifiziert werden kann, der einer angegebenen BTS-Aktivität oder einem angegebenen BTS-Prozess zugeordnet ist.

Anmerkung: Das Browse-Token sollte nur von dem Programm verwendet werden, das den Befehl STARTBROWSE absetzt.

Wenn Sie die Option ACTIVITYID oder PROCESS nicht angeben, untersucht CICS den Kontext (Kanal oder BTS) der Anforderung. Ist ein aktueller Kanal vorhanden, wird davon ausgegangen, dass es sich um einen Befehl STARTBROWSE CONTAINER (CHANNEL) handelt. Wenn eine aktuelle Aktivität vorhanden ist, werden die Container der Aktivität durchsucht. Wenn weder eine Aktivität noch ein Kanal vorhanden ist, wird eine Bedingung ACTIVITYERR 2 ausgelöst (siehe Beschreibung zur Bedingung ACTIVITYERR weiter unten).

Optionen

ACTIVITYID(datenwert)

Gibt die ID (1-52 Zeichen) der Aktivität an, deren Container durchsucht werden sollen.

In der Regel wurde die ID, die in dieser Option angegeben wird, in einem vorherigen Befehl GETNEXT ACTIVITY zurückgegeben.

BROWSETOKEN(datenbereich)

Gibt den Datenbereich (Vollwort-Binärwert) an, in den CICS das Browse-Token einfügt.

PROCESS(datenwert)

Gibt den Namen (1-36 Zeichen) des Prozesses an, dessen Container durchsucht werden sollen.

Anmerkung: Die Container, die dem Prozess zugeordnet sind (*Prozesscontainer*) sind im gesamten Prozess global verfügbar. Sie sind nicht mit den Containern der Stammaktivität identisch.

PROCESSTYPE(datenwert)

Gibt den Prozesstyp (1-8 Zeichen) des Prozesses an, der in der Option PROCESS angegeben wird.

Bedingungen

109 ACTIVITYERR

RESP2-Werte:

1

Die in der Option ACTIVITYID angegebene Aktivität konnte nicht gefunden werden.

2

Es wurde keine der Optionen ACTIVITYID, PROCESS oder CHANNEL angegeben und der Anforderung ist kein aktueller Kanal und keine aktuelle Aktivität zugeordnet.

29

Die Repository-Datei ist nicht verfügbar.

30

Es ist ein Ein-/Ausgabefehler für die Repository-Datei aufgetreten.

17 IOERR

RESP2-Werte:

30

Es ist ein Ein-/Ausgabefehler für die Repository-Datei aufgetreten.

70 NOTAUTH

RESP2-Werte:

101

Der Benutzer, der der absetzenden Task zugeordnet ist, ist nicht berechtigt, auf diese Ressource in der angeforderten Weise zuzugreifen.

108 PROCESSERR

RESP2-Werte:

3

Der in der Option PROCESS angegebene Prozess konnte nicht gefunden werden.

4

Der in der Option PROCESSTYPE angegebene Prozessstyp konnte nicht gefunden werden.

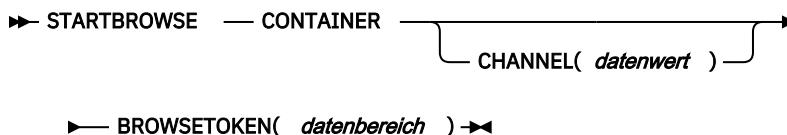
13

Die Anforderung hat das Zeitlimit überschritten. Es kann sein, dass die Beendigung einer anderen Task, die diesen Prozessdatensatz verwendet, verhindert wurde.

STARTBROWSE CONTAINER (CHANNEL)

Startet einen Suchvorgang für die Container, die einem Kanal zugeordnet sind.

STARTBROWSE CONTAINER



Bedingung: CHANNELERR

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl **STARTBROWSE CONTAINER (CHANNEL)** initialisiert ein Suchtoken, mit dem der Name jedes Datencontainers ermittelt werden kann, der einem angegebenen Kanal zugeordnet ist.

Anmerkung: Das Suchtoken sollte nur durch das Programm verwendet werden, das den Befehl **STARTBROWSE** ausgibt.

Falls Sie die Option CHANNEL nicht angeben, untersucht CICS den Kanal der Anforderung. Falls ein aktueller Kanal vorhanden ist, werden dessen Container durchsucht. Ist kein aktueller Kanal vorhanden, wird davon ausgegangen, dass es sich um einen Befehl „[STARTBROWSE CONTAINER \(BTS\)](#) auf Seite 591 handelt.

Die Reihenfolge, in der Container zurückgegeben werden, ist nicht definiert und kann sich ändern. Anwendungen sollten sich daher nicht auf die Reihenfolge der zurückgegebenen Container stützen. Falls Sie Anwendungen nutzen, die auf diese Weise geschrieben sind, finden Sie unter [Upgrade für Anwendungen](#) wichtige Hinweise. Angaben über die bewährten Verfahren bei der Verwendung eines Kanals in Anwendungen enthält der Abschnitt [Kanal entwerfen: Bewährte Verfahren](#).

Optionen

BROWSETOKEN(datenbereich)

Gibt einen Vollwort-Binärdatenbereich an, in den CICS das Suchtoken stellen soll.

CHANNEL(datenwert)

Gibt den 1 bis 16 Zeichen langen Namen des Kanals an, dessen Container durchsucht werden sollen. Hierbei muss es sich um den Namen entweder des aktuellen Kanals oder eines Kanals handeln, der durch das Programm erstellt wurde, von dem der Befehl **STARTBROWSE CONTAINER** ausgegeben wird. Sie können den Kanalnamen DFHTRANSACTION angeben, um den Transaktionskanal zu verwenden.

Falls diese Option nicht angegeben ist und ein Kanal den aktuellen Kontext darstellt, werden die Container des aktuellen Kanals durchsucht.

Bedingungen

122 CHANNELERR

RESP2-Werte:

2

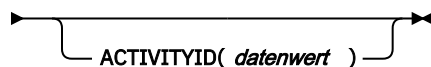
Der in der Option CHANNEL angegebene Kanal konnte nicht gefunden werden.

STARTBROWSE EVENT

Startet das Durchsuchen (Browsing) der Ereignisse, die einer BTS-Aktivität bekannt sind.

STARTBROWSE EVENT

➡ STARTBROWSE — EVENT — BROWSETOKEN(*datenbereich*) ➡



Bedingungen: ACTIVITYERR, INVREQ, IOERR, NOTAUTH

Beschreibung

Der Befehl STARTBROWSE EVENT initialisiert ein Browse-Token, mit dem jedes Ereignis (auch Unterereignis und Systemereignis) identifiziert werden kann, das sich im Gültigkeitsbereich einer angegebenen BTS-Aktivität befindet. Wenn Sie keine Aktivität angeben, werden Ereignisse im Gültigkeitsbereich der aktuellen Aktivität durchsucht.

Ein Durchsuchen, das durch einen Befehl STARTBROWSE EVENT gestartet wird, gibt folgende Typen von Ereignissen zurück:

- Atomare Ereignisse. Ein atomares Ereignis, das in diesem Befehl zurückgegeben wird, kann in das Prädikat eines zusammengesetzten Ereignisses einbezogen werden oder nicht, das heißt, es kann ein Unterereignis sein oder auch nicht.
- Zusammengesetzte Ereignisse.
- Systemereignisse.

Optionen

ACTIVITYID(datenwert)

Gibt die ID (1 - 52 Zeichen) der Aktivität an, deren Ereignisse durchsucht werden sollen.

Wenn Sie diese Option nicht angeben, werden Ereignisse durchsucht, die der aktuellen Aktivität bekannt sind.

BROWSETOKEN(datenbereich)

Gibt den Datenbereich (Vollwort-Binärwert) an, in den CICS das Browse-Token einfügt.

Bedingungen

109 ACTIVITYERR

RESP2-Werte:

1

Die in der Option ACTIVITYID angegebene Aktivitäts-ID bezieht sich auf keine Aktivität, die sich im Gültigkeitsbereich dieser Task befindet.

29

Die Repository-Datei ist nicht verfügbar.

30

Es ist ein Ein-/Ausgabefehler für die Repository-Datei aufgetreten.

16 INVREQ

RESP2-Werte:

1

Es ist keine aktuelle Aktivität im Bereich dieser Task vorhanden.

17 IOERR

RESP2-Werte:

30

Es ist ein Ein-/Ausgabefehler für die Repository-Datei aufgetreten.

70 NOTAUTH

RESP2-Werte:

101

Der Benutzer, der der absetzenden Task zugeordnet ist, ist nicht berechtigt, auf diese Ressource in der angeforderten Weise zuzugreifen.

STARTBROWSE PROCESS

Startet das Durchsuchen (Browsing) aller Prozesse eines angegebenen Typs im CICS-BTS-System.

STARTBROWSE PROCESS

➡ STARTBROWSE — PROCESS — PROCESSTYPE(*datenwert*) — BROWSETOKEN(*datenbereich*) ➡

Bedingungen: IOERR, NOTAUTH, PROCESSERR

Beschreibung

Der Befehl STARTBROWSE PROCESS initialisiert ein Browse-Token, mit dem jeder Prozess eines angegebenen Typs innerhalb des CICS-BTS-Systems identifiziert werden kann.

Wenn Sie dem BTS-System einen Prozess hinzufügen, verwenden Sie die Option PROCESSTYPE des Befehls DEFINE PROCESS, um den Prozess zu kategorisieren. Sie geben den Namen einer PROCESSTYPE-Ressourcendefinition an, die wiederum eine CICS-Dateidefinition benennt, die einer physischen VSAM-Datei (Repository) zugeordnet ist, in der Details des Prozesses und der zugehörigen Teilaktivitäten gespeichert werden. (Datensätze für mehrere Prozessstypen können in derselben Repository-Datei gespeichert werden.)

Mit dem Befehl STARTBROWSE PROCESS können Sie ein Durchsuchen eines Prozesses des angegebenen Typs starten.

Optionen

BROWSETOKEN(datenbereich)

Gibt den Datenbereich (Vollwort-Binärwert) an, in den CICS das Browse-Token einfügt.

PROCESSTYPE(datenwert)

Gibt den Prozesstyp (1 - 8 Zeichen) der Prozesse an, die durchsucht werden sollen.

Bedingungen

17 IOERR

RESP2-Werte:

29

Die Repository-Datei ist nicht verfügbar.

30

Es ist ein Ein-/Ausgabefehler für die Repository-Datei aufgetreten.

70 NOTAUTH

RESP2-Werte:

101

Der Benutzer, der der absetzenden Task zugeordnet ist, ist nicht berechtigt, auf diese Ressource in der angeforderten Weise zuzugreifen.

108 PROCESSERR

RESP2-Werte:

1

Es konnten keine Prozesse dieses Prozesstyps gefunden werden.

4

Der in der Option PROCESSTYPE angegebene Prozesstyp konnte nicht gefunden werden.

13

Die Anforderung hat das Zeitlimit überschritten. Es kann sein, dass die Beendigung einer anderen Task, die diesen Prozessdatensatz verwendet, verhindert wurde.

STARTBROWSE TIMER

Startet einen Suchvorgang für einen BTS-Zeitgeber.

STARTBROWSE TIMER



Bedingungen: ACTIVITYERR, INVREQ, IOERR, NOTAUTH, TIMERERR

Beschreibung

Der Befehl **STARTBROWSE TIMER** initialisiert ein Suchtoken, mit dem ein BTS-Zeitgeber ermittelt werden kann.

Optionen

ACTIVITYID(datenwert)

Gibt die ID (1 bis 52 Zeichen) der Aktivität an, der der Zeitgeber zugeordnet ist.

Falls diese Option nicht angegeben ist, wird von der aktuellen Aktivität ausgegangen.

BROWSETOKEN(datenbereich)

Gibt einen Datenbereich als Vollwort-Binärzahl an, in den CICS das Suchtoken schreiben soll.

TIMER(datenwert)

Gibt den Namen (1 bis 16 Zeichen) des Zeitgebers an.

Bedingungen**109 ACTIVITYERR**

RESP2-Werte:

3

Die in der Option ACTIVITYID angegebene Aktivität konnte nicht gefunden werden.

29

Die Repository-Datei ist nicht verfügbar.

30

Es ist ein Ein-/Ausgabefehler für die Repository-Datei aufgetreten.

16 INVREQ

RESP2-Werte:

1

Der Befehl wurde außerhalb des Gültigkeitsbereichs einer momentan aktiven Aktivität abgesetzt.

17 IOERR

RESP2-Werte:

30

Es ist ein Ein-/Ausgabefehler für die Repository-Datei aufgetreten.

70 NOTAUTH

RESP2-Werte:

101

Der Benutzer, der der ausgebenden Task zugeordnet ist, ist nicht berechtigt, auf die angeforderte Weise auf diese Ressource zuzugreifen.

115 TIMERERR

RESP2-Werte:

1

Der in der Option TIMER angegebene Zeitgeber konnte nicht gefunden werden.

SUSPEND

Setzt eine Task aus.

SUSPEND

➡ SUSPEND →

Dieser Befehl ist threadsicher.

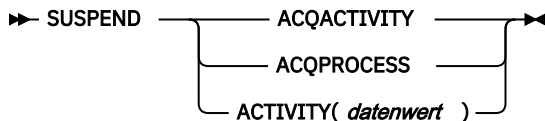
Beschreibung

Der Befehl SUSPEND gibt die Steuerung an eine Task mit höherer oder gleicher Zuteilungspriorität frei. Die Steuerung wird hierbei an die Task, die den Befehl ausgibt, freigegeben, sobald keine andere Task mit einer höheren oder gleichen Priorität verarbeitungsbereit ist.

SUSPEND (BTS)

Setzt einen BTS-Prozess oder eine BTS-Aktivität aus.

SUSPEND (BTS)



Bedingungen: ACTIVITYBUSY, ACTIVITYERR, INVREQ, IOERR, LOCKED, PROCESSERR

Beschreibung

Der Befehl SUSPEND (BTS) verhindert, dass ein BTS-Prozess oder eine BTS-Aktivität erneut angeschlossen (reaktiviert) wird, wenn Ereignisse im zugehörigen Ereignispool ausgelöst werden.

Der einzige Prozess, den ein Programm aussetzen kann, ist derjenige, den es in der aktuellen Arbeitseinheit angefordert hat.

Die einzigen Aktivitäten, die ein Programm aussetzen kann, sind die folgenden:

- Seine eigenen untergeordneten Aktivitäten, falls es als Aktivierung einer Aktivität ausgeführt wird. Es kann mehrere der eigenen untergeordneten Aktivitäten innerhalb derselben Arbeitseinheit aussetzen.
- Die Aktivität, die es durch einen Befehl ACQUIRE ACTIVITYID in der aktuellen Arbeitseinheit angefordert hat.

Zur Fortsetzung eines ausgesetzten Prozesses oder einer ausgesetzten Aktivität muss ein Befehl RESUME abgesetzt werden.

Optionen

ACQACTIVITY

Gibt an, dass die auszusetzende Aktivität diejenige ist, die von der aktuellen Arbeitseinheit durch einen Befehl ACQUIRE ACTIVITYID angefordert wurde.

ACQPROCESS

Gibt an, dass der Prozess, der zurzeit durch den Requester angefordert ist, ausgesetzt werden soll.

ACTIVITY(datenwert)

Gibt den Namen (1 - 16 Zeichen) der untergeordneten Aktivität an, die ausgesetzt werden soll.

Bedingungen

107 ACTIVITYBUSY

RESP2-Werte:

19

Die Anforderung hat das Zeitlimit überschritten. Es kann sein, dass die Beendigung einer anderen Task, die diesen Aktivitätsdatensatz verwendet, verhindert wurde.

109 ACTIVITYERR

RESP2-Werte:

8

Die in der Option ACTIVITY angegebene Aktivität konnte nicht gefunden werden.

16 INVREQ

RESP2-Werte:

4

Mit der Option ACTIVITY wurde eine untergeordnete Aktivität angegeben, aber der Befehl wurde außerhalb des Geltungsbereichs einer gegenwärtig aktiven Aktivität ausgegeben.

14

Die Aktivität ist im Modus COMPLETE oder CANCELLING und kann daher nicht ausgesetzt werden.

15

Die Option ACQPROCESS wurde verwendet, jedoch hat die Arbeitseinheit, die die Anforderung abgesetzt hat, keinen Prozess angefordert.

24

Die Option ACQACTIVITY wurde verwendet, jedoch hat die Arbeitseinheit, die die Anforderung abgesetzt hat, keine Aktivität angefordert.

17 IOERR

RESP2-Werte:

29

Die Repository-Datei ist nicht verfügbar.

30

Es ist ein Ein-/Ausgabefehler für die Repository-Datei aufgetreten.

100 LOCKED

Die Anforderung kann nicht ausgeführt werden, weil eine Sperre für den relevanten Datensatz in der Repository-Datei besteht.

108 PROCESSERR

RESP2-Werte:

5

Der Prozess konnte nicht gefunden werden.

SYNCPOINT

Erstellt einen Synchronisationspunkt.

SYNCPOINT

➡ SYNCPOINT ➡

Bedingungen: INVREQ, ROLLEDBACK

Dieser Befehl ist threadsicher.

Anmerkung: Der Wiederherstellungsmanager verarbeitet diesen Befehl nach Möglichkeit für einen offenen Tasksteuerblock, um den Wechsel von Tasksteuerblöcken zu minimieren. Die Synchronisationspunktverarbeitung kann für einen offenen Tasksteuerblock für alle als threadsicher deklarierten Ressourcentypen stattfinden, auf die in der Arbeitseinheit zugegriffen wurde. Falls in der Arbeitseinheit auf nicht als threadsicher deklarierte Ressourcentypen zugegriffen wurde, wechselt der Wiederherstellungsmanager für solche Ressourcentypen zum QR-Tasksteuerblock. Ein CICS-Ressourcentyp deklariert sich selbst beim Wiederherstellungsmanager als threadsicher, falls die CICS-Befehle für den Ressourcentyp threadsicher sind.

Beschreibung

Der Befehl SYNCPOINT unterteilt eine Task (normalerweise mit langer Laufzeit) in kleinere Arbeitseinheiten. Er gibt an, dass alle Änderungen, die durch die Task seit ihrem letzten Synchronisationspunkt an wiederherstellbaren Ressourcen vorgenommen wurden, festgeschrieben werden sollen.

Anmerkung: Falls während der Festschreibungsphase (Phase 2) der Synchronisationspunktverarbeitung ein Fehler auftritt, wird keine Fehlerbedingung zurückgegeben und die Transaktion nicht abnormal beendet. Nachfolgende Arbeitseinheiten in der Transaktion können normal fortgesetzt werden. Weitere Informationen finden Sie unter Wiederherstellungs- und Absturzbehandlung bei Arbeitseinheiten.

Bedingungen

16 INVREQ

RESP2-Werte:

200

Der Befehl SYNCPOINT wurde in einem Programm ausgegeben, zu dem von einem fernen System aus ohne Angabe der Option SYNCONRETURN eine Verbindung hergestellt wurde, oder es wurde eine lokale Verbindung zum Befehl hergestellt, die mit EXECUTIONSET=DPLSUBSET definiert ist.

Standardaktion: Die Task wird abnormal beendet.

82 ROLLEDBACK

Tritt auf, wenn ein Befehl SYNCPOINT durch ein fernes System, das den Synchronisationspunkt nicht festschreiben kann, zu einem Rollback veranlasst wird. Alle Änderungen, die in der aktuellen Arbeitseinheit an wiederherstellbaren Ressourcen vorgenommen wurden, werden zurückgesetzt.

Standardaktion: Die Task wird abnormal beendet.

SYNCPOINT ROLLBACK

Führt eine Zurücksetzung auf den letzten Synchronisationspunkt durch.

SYNCPOINT ROLLBACK

➡ SYNCPOINT — ROLLBACK ➡

Bedingung: INVREQ

Dieser Befehl ist threadsicher.

Anmerkung: Der Wiederherstellungsmanager verarbeitet diesen Befehl nach Möglichkeit für einen offenen Tasksteuerblock, um den Wechsel von Tasksteuerblöcken zu minimieren. Die Synchronisationspunktverarbeitung kann für einen offenen Tasksteuerblock für alle als threadsicher deklarierten Ressourcentypen stattfinden, auf die in der Arbeitseinheit zugegriffen wurde. Falls in der Arbeitseinheit auf nicht als threadsicher deklarierte Ressourcentypen zugegriffen wurde, wechselt der Wiederherstellungsmanager für solche Ressourcentypen zum QR-Tasksteuerblock. Ein CICS-Ressourcentyp deklariert sich selbst beim Wiederherstellungsmanager als threadsicher, falls die CICS-Befehle für den Ressourcentyp threadsicher sind.

Optionen

ROLLBACK

Gibt an, dass alle Änderungen, die durch die Task seit ihrem letzten Synchronisationspunkt an wiederherstellbaren Ressourcen vorgenommen wurden, zurückgesetzt werden sollen.

Mit dieser Option können beispielsweise eine Routine HANDLE ABEND bereinigt oder Datenbankänderungen widerrufen werden, nachdem das Anwendungsprogramm nicht behebbare Fehler in seinen Eingabedaten festgestellt hat.

Falls die Arbeitseinheit ferne wiederherstellbare Ressourcen unter Verwendung einer MRO- oder APPC-Sitzung aktualisiert, wird die Option ROLLBACK an die Back-End-Transaktion weitergegeben.

Wenn ein Dialog für die verteilte Transaktionsverarbeitung verwendet wird, sind für das ferne Anwendungsprogramm die EIB-Felder EIBSYNRB, EIBERR und EIBERRCD festgelegt. Damit der Dialog fortgesetzt werden kann, sollte das ferne Anwendungsprogramm einen Befehl SYNCPOINT ROLLBACK ausführen.

Wenn die Spiegeltransaktion in die Arbeitseinheit unter Verwendung einer MRO- oder APPC-Sitzung einbezogen ist, berücksichtigt der Spiegel die Rollbackanforderung, widerruft Änderungen und wird anschließend normal beendet.

Diese Option wird bei z/OS Communications Server-Sitzungen mit LU-Typ 6.1 für die Spiegelung oder Back-End-Transaktionen nicht unterstützt. In diesen Fällen könnten Front-End-Transaktionen abgebrochen werden, um das Zurücksetzen von Back-End-Transaktionen zu bewirken.

Anmerkung: Falls während der Zurücksetzungsphase (Phase 2) der Synchronisationspunktverarbeitung ein Fehler auftritt, wird keine Fehlerbedingung zurückgegeben und die Transaktion nicht abnormal beendet. Nachfolgende Arbeitseinheiten in der Transaktion können normal fortgesetzt werden. Weitere Informationen finden Sie unter Wiederherstellungs- und Absturzbehandlung bei Arbeitseinheiten.

Anmerkung: Eine verzögerte EXEC CICS-Anforderung SEND wird während eines Befehls **SYNCPOINT ROLLBACK** abgebrochen.

Bedingungen

16 INVREQ

RESP2-Werte:

200

Der Befehl SYNCPOINT ROLLBACK wurde in einem Programm ausgegeben, zu dem von einem fernen System aus ohne Angabe der Option SYNCONRETURN eine Verbindung hergestellt wurde, oder es wurde eine lokale Verbindung zum Befehl hergestellt, die mit EXECUTIONSET=DPLSUBSET definiert ist.

Standardaktion: Die Task wird abnormal beendet.

TEST EVENT

Testet, ob ein BTS-Ereignis ausgelöst wurde.

TEST EVENT

► TEST — EVENT(*datenwert*) — FIRESTATUS(*cvda*) ◄

Bedingungen: EVENTERR, INVREQ

Beschreibung

Der Befehl TEST EVENT testet, ob ein angegebenes BTS-Ereignis aufgetreten ist (ausgelöst wurde).

Optionen

EVENT(datenwert)

Gibt den Namen (1 - 16 Zeichen) des Ereignisses an, dessen Auftreten getestet werden soll.

FIRESTATUS(cvda)

Die Option FIRESTATUS gibt den Ausführungsstatus des Ereignisses zurück. Die CVDA-Werte lauten wie folgt:

FIRED

Das Ereignis wurde ausgelöst.

NOTFIRED

Das Ereignis wurde nicht ausgelöst.

Bedingungen

111 EVENTERR

RESP2-Werte:

4

Das in der Option EVENT angegebene Ereignis wird von BTS nicht erkannt.

16 INVREQ

RESP2-Werte:

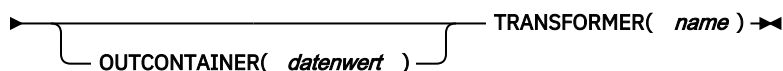
Der Befehl wurde außerhalb des Gültigkeitsbereichs einer Aktivität abgesetzt.

TRANSFORM DATATOJSON

Mit dem Befehl **TRANSFORM DATATOJSON** können Sie Anwendungsdaten in JSON konvertieren.

TRANSFORM DATATOJSON

➤ TRANSFORM DATATOJSON — CHANNEL(*datenwert*) — INCONTAINER(*datenwert*) ➤



Bedingungen: CHANNELERR, CONTAINERERR, INVREQ, NOTFND

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl **TRANSFORM DATATOJSON** wandelt Anwendungsdaten in JSON um und verwendet hierbei Zuordnungen, die in der JSON-Bindung definiert sind. Die Ressource für das JSON-Umsetzungsprogramm ist ein Paket, das definiert, wo sich die JSON-Bindung und das JSON-Schema befinden.

Optionen

CHANNEL(*datenwert*)

Geben Sie den Namen des Kanals an, der zur Übergabe der Container verwendet werden soll, in denen sich die in den Optionen CONTAINER und INCONTAINER angegebenen Daten befinden. Der Name des Kanals muss 16 Zeichen lang sein. Falls der Kanalname kürzer als 16 Zeichen ist, müssen Sie den Datenwert mit nachgestellten Leerzeichen auffüllen. Sie können DFHTRANSACTION angeben, damit der Transaktionskanal verwendet wird.

INCONTAINER(*datenwert*)

Geben Sie den Namen des Containers an, der die zu konvertierenden Anwendungsdaten enthält. Dieser Container muss im Kanal vorhanden sein. Der Name des Containers muss 16 Zeichen lang sein. Falls der Containername kürzer als 16 Zeichen ist, müssen Sie den Datenwert mit nachgestellten Leerzeichen auffüllen.

CICS liest Daten aus diesem Container im Bitmodus (BIT) ein.

OUTCONTAINER(*datenwert*)

Geben Sie optional den Namen des Containers an, der nach Abschluss des Befehls die JSON-Ausgabe enthalten soll. Der Name des Containers muss 16 Zeichen lang sein. Falls der Containername kürzer als 16 Zeichen ist, müssen Sie den Datenwert mit nachgestellten Leerzeichen auffüllen. Falls diese Option nicht angegeben ist, wird DFHJSON-JSJSON verwendet.

Erstellen Sie vor Ausgabe des Befehls nicht den Zielcontainer, da durch den Befehl selbst ein Container erstellt und gefüllt wird. Falls der Container vorhanden ist, wird er gelöscht, bevor die Umwandlung stattfindet. CICS füllt diesen Container im Zeichenmodus (CHAR).

TRANSFORMER(*name*)

Geben Sie den 16 Byte umfassenden Namen des Ressourcenpakets für das JSON-Umsetzungsprogramm an, das CICS zum Umwandeln der Daten in JSON verwendet. Die Ressource definiert die JSON-Bindung und das JSON-Schema. Der Name der Ressource für das JSON-Umsetzungsprogramm muss 16 Zeichen lang sein. Falls der Ressourcenname kürzer als 16 Zeichen ist, müssen Sie den Wert mit Leerzeichen auffüllen.

Damit der Befehl ordnungsgemäß ausgeführt wird, muss dieses Ressourcenpaket installiert und aktiviert sein.

Bedingungen

122 CHANNELERR

RESP2-Werte:

- 1
Der durch den Parameter **CHANNEL** angegebene Kanal ist falsch.
- 2
Der durch den Parameter **CHANNEL** angegebene Kanal wurde nicht gefunden.

110 CONTAINERERR

RESP2-Werte:

- 3
Der durch den Parameter **INCONTAINER** angegebene Container wurde nicht gefunden.

16 INVREQ

RESP2-Werte:

- 1
Die Paketressource für das JSON-Umsetzungsprogramm ist nicht aktiviert.
- 6
Die Anwendungsdaten können nicht konvertiert werden. Weitere Informationen finden Sie in der Fehlernachricht im Container DFHJSON-ERRORMSG.
- 8
Der Container INCONTAINER ist nicht mit Daten im Bitmodus (BIT) gefüllt.
- 101
Die Benutzer-ID besitzt keine Berechtigung für die Verwendung des JSON-Umsetzungsprogramms.

13 NOTFND

RESP2-Werte:

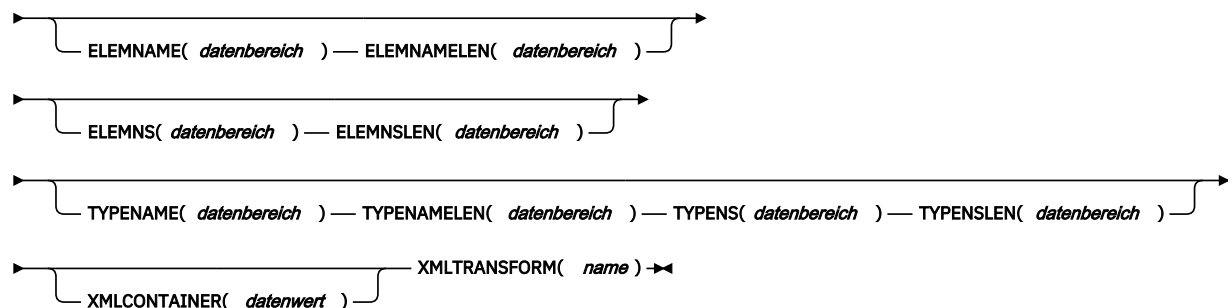
- 1
Die Paketressource für das JSON-Umsetzungsprogramm wurde nicht gefunden.

TRANSFORM DATATOXML

Mit dem Befehl **TRANSFORM DATATOXML** können Sie Anwendungsdaten in XML konvertieren.

TRANSFORM DATATOXML

►► TRANSFORM DATATOXML — CHANNEL(*datenwert*) — DATCONTAINER(*datenwert*) —►



Bedingungen: NOTFND, LENGERR, CHANNELERR, CONTAINERERR, INVREQ

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl **TRANSFORM DATATOXML** wandelt Anwendungsdaten in XML um und verwendet hierbei Zuordnungen, die in der XML-Bindung definiert sind. Die XMLTRANSFORM-Ressource definiert, wo sich die XML-Bindung und das XML-Schema befinden.

Optionen

CHANNEL (datenwert)

Geben Sie den Namen des Kanals an, der zur Übergabe der Container verwendet werden soll, in denen sich die in den Optionen XMLCONTAINER und DATCONTAINER angegebenen Daten befinden. Der Name des Kanals muss 16 Zeichen lang sein. Falls der Kanalname kürzer als 16 Zeichen ist, müssen Sie den Datenwert mit nachgestellten Leerzeichen auffüllen. Sie können den Kanalnamen DFHTRANSACTION angeben, um den Transaktionskanal zu verwenden.

DATCONTAINER (datenwert)

Geben Sie den Namen des Containers an, der die zu konvertierenden Anwendungsdaten enthält. Dieser Container muss im Kanal vorhanden sein. Der Name des Containers muss 16 Zeichen lang sein. Falls der Containername kürzer als 16 Zeichen ist, müssen Sie den Datenwert mit nachgestellten Leerzeichen auffüllen.

CICS liest Daten aus diesem Container im Bitmodus (BIT) ein.

ELEMNAME (datenbereich)

Geben Sie den Namen eines XML-Elements an. CICS gibt den lokalen Namen des XML-Elements zurück, der von CICS generiert wird.

ELEMNAMELEN (datenbereich)

Geben Sie die Länge des XML-Elements in der Option ELEMNAME als Vollwort-Binärzahl an. Der Maximalwert von ELEMNAMELEN ist 255.

ELEMNS (datenbereich)

Geben Sie Namensbereichs-URI des XML-Elements an, das in der Option ELEMNAME zurückgegeben wird.

ELEMNSLEN (datenbereich)

Geben Sie die Länge des Namensbereichs der Option ELEMNS als Vollwort-Binärzahl an. Der Maximalwert von ELEMNSLEN ist 255.

TYPENAME (datenbereich)

Geben Sie den Wert des Attributs 'xsi:type' für das XML-Element an, das in der Option ELEMNAME zurückgegeben wird.

TYPENAMELEN (datenbereich)

Geben Sie die Länge des Wertes für das Attribut 'xsi:type', das in der Option TYPENAME zurückgegeben wird, als Vollwort-Binärzahl an.

TYPENS (datenbereich)

Geben Sie den Namensbereich des Attributs 'xsi:type' für das XML-Element an, das in der Option ELEMNAME zurückgegeben wird.

TYPENSLEN (datenbereich)

Geben Sie die Länge des Namensbereichs für das Attribut 'xsi:type', das in der Option TYPENS zurückgegeben wird, als Vollwort-Binärzahl an.

XMLCONTAINER (datenwert)

Geben Sie den Namen des Containers an, der nach Abschluss des Befehls die XML-Ausgabe enthält. Der Name des Containers muss 16 Zeichen lang sein. Falls der Containername kürzer als 16 Zeichen ist, müssen Sie den Datenwert mit nachgestellten Leerzeichen auffüllen.

Sie müssen den Zielcontainer nicht erstellen, bevor Sie den Befehl ausgeben. Der Container wird durch den Befehl selbst erstellt und gefüllt. Falls der Container vorhanden ist und mit dem Datentyp BIT definiert wurde, wird er im Rahmen der Befehlsverarbeitung gelöscht und mit dem Typ CHAR erneut erstellt.

XMLTRANSFORM (datenwert)

Geben Sie den Namen der XMLTRANSFORM-Ressource an, die CICS zum Umwandeln der Daten in XML verwendet. Die Ressource definiert die XML-Bindung und das XML-Schema. Der Name der XMLTRANSFORM-Ressource muss 32 Zeichen lang sein. Falls der Ressourcename kürzer als 32 Zeichen ist, müssen Sie den Wert mit Leerzeichen auffüllen.

Bedingungen

13 NOTFND

RESP2-Werte:

1

Die XMLTRANSFORM-Ressource wurde nicht gefunden.

122 CHANNELERR

RESP2-Werte:

1

Der durch den Parameter **CHANNEL** angegebene Kanal ist falsch.

2

Der durch den Parameter **CHANNEL** angegebene Kanal wurde nicht gefunden.

110 CONTAINERERR

RESP2-Werte:

1

Der durch den Parameter **XMLCONTAINER** angegebene Container wurde nicht gefunden.

2

Der durch den Parameter **NSCONTAINER** angegebene Container wurde nicht gefunden.

3

Der durch den Parameter **DATCONTAINER** angegebene Container wurde nicht gefunden.

22 LENGERR

RESP2-Werte:

1

Die Daten in dem durch den Parameter **DATCONTAINER** angegebenen Container sind zu kurz für den angegebenen Umwandlungstyp.

2

Der Puffer für den Wert in **ELEMNAME** ist zu klein.

3

Der Puffer für den Wert in **ELEMNS** ist zu klein.

4

Der Puffer für den Wert in **TYPENAME** ist zu klein.

5

Der Puffer für den Wert in **TYPENS** ist zu klein.

6

Der Wert von **ELEMNAMELEN** überschreitet den Maximalwert von 255.

7

Der Wert von **ELEMNSLEN** überschreitet den Maximalwert von 255.

16 INVREQ

RESP2-Werte:

1

Die XMLTRANSFORM-Ressource ist nicht aktiviert.

2

Der mit **XMLCONTAINER** angegebene Container ist leer.

3

Die XML-Eingabedaten sind ungültig. Weitere Informationen enthält die Fehlermeldung im Container DFH-XML-ERRORMSG.

4

Die Eingabedaten können nicht konvertiert werden. Weitere Informationen enthält die Fehlermeldung im Container DFH-XML-ERRORMSG.

- 5** Die Anwendungsdaten sind ungültig. Weitere Informationen enthält die Fehlnachricht im Container DFH-XML-ERRORMSG.
- 6** Die Anwendungsdaten können nicht konvertiert werden. Weitere Informationen enthält die Fehlnachricht im Container DFH-XML-ERRORMSG.
- 8** Der Anwendungsdatencontainer wurde nicht im Bitmodus (BIT) gefüllt.
- 9** Die XMLTRANSFORM-Ressource unterstützt das angeforderte XML-Element nicht.
- 10** Die XMLTRANSFORM-Ressource unterstützt den angeforderten XML-Typ nicht.
- 11** Beim Herstellen der Verbindung zu einem vom Anbieter bereitgestellten Umsetzungsprogramms trat ein Problem auf.
- 13** Der Parameter **CHANNEL** wurde nicht angegeben und ist erforderlich.
- 14** Der Parameter **ELEMNAME** wurde nicht angegeben und ist erforderlich.
- 15** Der Parameter **ELEMNS** wurde nicht angegeben und ist erforderlich.
- 16** Der Parameter **DATCONTAINER** wurde nicht festgelegt und ist erforderlich.
- 17** Bei der Laufzeitvalidierung trat ein Fehler auf.
- 18** Hinsichtlich des Datentyps für den Container trat ein Fehler auf.
- 101** Der Benutzer ist nicht zur Verwendung der XMLTRANSFORM-Ressource berechtigt.

TRANSFORM JSONTODATA

Mit dem Befehl **TRANSFORM JSONTODATA** können Sie JSON in Anwendungsdaten konvertieren.

TRANSFORM JSONTODATA

►► TRANSFORM JSONTODATA — CHANNEL(*datenwert*) — INCONTAINER(*datenwert*) —►

┌──────────────────────────────────┐ TRANSFORMER(*name*) ─►
└──────────────────────────────────┘ OUTCONTAINER(*datenwert*) ─

Bedingungen: CHANNELERR, CONTAINERERR, INVREQ, NOTFND

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl **TRANSFORM JSONTODATA** wandelt JSON in Anwendungsdaten um. Die Ressource für das JSON-Umsetzungsprogramm ist ein Paket, das definiert, wo sich die JSON-Bindung und das JSON-Schema für die Umwandlung von JSON in Anwendungsdaten befinden.

Optionen

CHANNEL (*datenwert*)

Geben Sie den Namen des Kanals an, der zur Übergabe der Container verwendet werden soll, in denen sich die in den Optionen CONTAINER und INCONTAINER angegebenen Daten befinden. Der Name des Kanals muss 16 Zeichen lang sein. Falls der Kanalname kürzer als 16 Zeichen ist, müssen Sie den Datenwert mit nachgestellten Leerzeichen auffüllen. Sie können DFHTRANSACTION angeben, damit der Transaktionskanal verwendet wird.

INCONTAINER (*datenwert*)

Geben Sie den 16 Byte umfassenden Namen des Containers an, der die zu konvertierenden JSON-Daten enthält. Dieser Container muss bereits vorhanden und im Zeichenmodus (CHAR) gefüllt worden sein. Falls er im Bitmodus (BIT) gefüllt wurde, versucht CICS, die Datencodierung festzustellen.

OUTCONTAINER (*datenwert*)

Geben Sie optional den Namen des Containers an, der die konvertierten Daten enthalten soll. Der Name des Containers muss 16 Zeichen lang sein. Falls der Containername kürzer als 16 Zeichen ist, müssen Sie den Datenwert mit nachgestellten Leerzeichen auffüllen. Falls diese Option nicht angegeben ist, wird DFHJSON-DATA verwendet.

Erstellen Sie vor Ausgabe des Befehls nicht den Zielcontainer, da durch den Befehl selbst ein Container erstellt und gefüllt wird. Falls der Container vorhanden ist, wird er gelöscht, bevor die Umwandlung stattfindet. CICS füllt diesen Container im Bitmodus (BIT).

TRANSFORMER (*name*)

Geben Sie den 16 Byte umfassenden Namen des Ressourcenpakets für das JSON-Umsetzungsprogramm an, das CICS zum Umwandeln der Daten in JSON verwendet. Die Ressource definiert die JSON-Bindung und das JSON-Schema. Der Name der Ressource für das JSON-Umsetzungsprogramm muss 16 Zeichen lang sein. Falls der Ressourcenname kürzer als 16 Zeichen ist, müssen Sie den Wert mit Leerzeichen auffüllen.

Bedingungen

122 CHANNELERR

RESP2-Werte:

1

Der durch den Parameter **CHANNEL** angegebene Kanal ist falsch.

2

Der durch den Parameter **CHANNEL** angegebene Kanal wurde nicht gefunden.

110 CONTAINERERR

RESP2-Werte:

1

Der durch den Parameter **INCONTAINER** angegebene Container wurde nicht gefunden.

16 INVREQ

RESP2-Werte:

1

Die Paketressource für das JSON-Umsetzungsprogramm ist nicht aktiviert.

4

Die Anwendungsdaten können nicht konvertiert werden. Weitere Informationen finden Sie in der Fehlnachricht im Container DFHJSON-ERRORMSG.

7

Der Container INCONTAINER ist nicht mit Daten im Zeichenmodus (CHAR) gefüllt.

101

Die Benutzer-ID besitzt keine Berechtigung für die Verwendung des JSON-Umsetzungsprogramms.

13 NOTFND

RESP2-Werte:

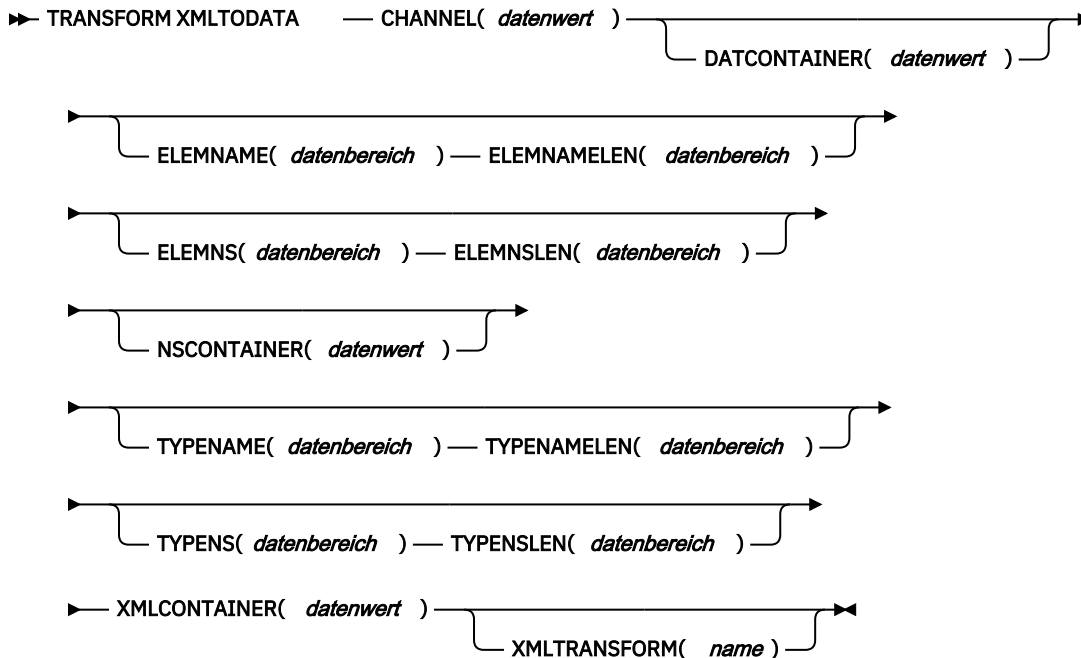
1

Die Paketressource für das JSON-Umsetzungsprogramm wurde nicht gefunden.

TRANSFORM XMLTODATA

Mit dem Befehl **TRANSFORM XMLTODATA** können Sie XML-Daten in Anwendungsdaten konvertieren.

TRANSFORM XMLTODATA



Bedingungen: CHANNELERR, CONTAINERERR, INVREQ, LENGERR, NOTFND,

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl **TRANSFORM XMLTODATA** kann entweder XML-Daten in Anwendungsdaten konvertieren oder XML-Daten abfragen, um Informationen zu den XML-Elementen an das Anwendungsprogramm zurückzugeben. Die XMLTRANSFORM-Ressource definiert die Position der XML-Bindung und des XML-Schemas für die Umwandlung der XML-Daten in Anwendungsdaten. Falls Sie im Befehl keine XMLTRANSFORM-Ressource angeben, fragt CICS stattdessen die XML-Daten ab.

Optionen

CHANNEL(*datenwert*)

Geben Sie den 16 Byte umfassenden Namen des Kanals an, der zur Übergabe der Container verwendet werden soll, in denen sich die in den Optionen XMLCONTAINER und DATCONTAINER angegebenen Daten befinden. Der Name des Kanals muss 16 Zeichen lang sein. Falls der Kanalname kürzer als 16 Zeichen ist, müssen Sie den Datenwert mit nachgestellten Leerzeichen auffüllen. Sie können den Kanalnamen DFHTRANSACTION angeben, um den Transaktionskanal zu verwenden.

DATCONTAINER(*datenwert*)

Geben Sie den 16 Byte umfassenden Namen des Ausgabecontainers an, den CICS mit den konvertierten Daten füllt. Der Name des Containers muss 16 Zeichen lang sein. Falls der Containername kürzer als 16 Zeichen ist, müssen Sie den Datenwert mit nachgestellten Leerzeichen auffüllen.

CICS füllt diesen Container im Bitmodus (BIT).

ELEMNAME (datenbereich)

Geben Sie einen Eingabewert ein, um den Namen eines XML-Elements zurückzugeben. CICS füllt die Option ELEMNAME mit dem lokalen Namen des ersten XML-Elements, das im Container XMLCONTAINER gefunden wird. Die Anwendung muss ebenfalls einen Eingabewert für die Option ELEMNAMELEN angeben, die die maximale Länge des Datenbereichs festlegt.

ELEMNAMELEN (datenbereich)

Geben Sie einen Eingabewert an, damit die Länge des XML-Elements in der Option ELEMNAME als Vollwort-Binärzahl zurückgegeben wird. CICS aktualisiert den Wert der Option ELEMNAMELEN, um die tatsächliche Länge des gefundenen Elementnamens anzugeben.

ELEMNS (datenbereich)

Geben Sie einen Eingabewert an, damit die Länge der Namensbereichs-URI für das XML-Element zurückgegeben wird, auf das sich die Option ELEMNAME bezieht. CICS füllt diesen Parameter mit der Namensbereichs-URI des ersten XML-Elements, das im Container XMLCONTAINER gefunden wird. Die Anwendung muss ebenfalls einen Eingabewert für die Option ELEMNSLEN angeben, die die maximale Länge des Datenbereichs festlegt.

ELEMNSLEN (datenbereich)

Geben Sie einen Eingabewert an, damit die Länge der Option ELEMNS zurückgegeben wird. CICS aktualisiert den Wert der Option ELEMNSLEN, um die tatsächliche Länge der Namensbereichs-URI anzugeben.

NSCONTAINER (datenwert)

Geben Sie den 16 Byte umfassenden Namen des Containers an, der eine Liste mit geltenden XML-Namensbereichsdeklarationen enthält. Diese XML-Namensbereichsdeklarationen können im Hauptteil des Containers XMLCONTAINER referenziert werden. Der Container muss im Zeichenmodus (CHAR) gefüllt sein.

TYPENAME (datenbereich)

Geben Sie einen Eingabewert an, damit das Attribut `xml:type` für das XML-Element zurückgegeben wird, auf das sich die Option ELEMNAME bezieht. Dieser Parameter wird von CICS mit dem lokalen Namen des Attributs `xml:type` für den ersten XML-Tag gefüllt, der im Container XMLCONTAINER gefunden wird. Falls der erste XML-Tag kein Attribut `xml:type` besitzt, bleibt dieser Parameter leer. Falls die Anwendung einen Wert für die Option TYPENAME angibt, überschreibt dieser Wert alle Element- und Typinformationen in den bereitgestellten XML-Daten und CICS versucht, die Umwandlung unter Verwendung des Typnamens durchzuführen, der durch diese Anwendung (zusammen mit der Option TYPENS) bereitgestellt wird.

TYPENAMELEN (datenbereich)

Geben Sie einen Eingabewert an, damit die Länge der Option TYPENAME zurückgegeben wird.

TYPENS (datenbereich)

Geben Sie einen Eingabewert an, damit der Namensbereich des Attributs `xml:type` für das XML-Element zurückgegeben wird, auf das sich die Option ELEMNAME bezieht. Dieser Parameter wird von CICS mit dem Namensbereich des Attributs `xml:type` für das erste XML-Element gefüllt, das im Container XMLCONTAINER gefunden wird.

TYPENSLEN (datenbereich)

Geben Sie einen Eingabewert an, damit die Länge der Option TYPENS zurückgegeben wird.

XMLCONTAINER (datenwert)

Geben Sie den 16 Byte umfassenden Namen des Eingabecontainers an, der die zu konvertierenden XML-Daten enthält. Dieser Container muss bereits vorhanden und im Zeichenmodus (CHAR) gefüllt worden sein. Falls er im Bitmodus (BIT) gefüllt wurde, versucht CICS, die Datencodierung festzustellen.

XMLTRANSFORM (name)

Geben Sie den 32 Byte umfassenden Namen der XMLTRANSFORM-Ressource an, die CICS zum Umwandeln der Daten in XML verwendet. Die Ressource definiert die XML-Bindung und das XML-Schema. Der Name der XMLTRANSFORM-Ressource muss 32 Zeichen lang sein. Falls der Ressourcename kürzer als 32 Zeichen ist, müssen Sie den Wert mit Leerzeichen auffüllen.

Falls Sie die Option XMLTRANSFORM nicht angeben, findet keine Datenkonvertierung statt. Stattdessen fragt die Anwendung die XML-Daten ab. CICS gibt die Informationen zum XML-Element und zum Typ in den Optionen ELEMNAME, ELEMNS, TYPENAME und TYPENS zurück.

Bedingungen

13 NOTFND

RESP2-Werte:

1

Die XMLTRANSFORM-Ressource wurde nicht gefunden.

122 CHANNELERR

RESP2-Werte:

1

Der durch den Parameter **CHANNEL** angegebene Kanal ist falsch.

2

Der durch den Parameter **CHANNEL** angegebene Kanal wurde nicht gefunden.

110 CONTAINERERR

RESP2-Werte:

1

Der durch den Parameter **XMLCONTAINER** angegebene Container wurde nicht gefunden.

2

Der durch den Parameter **NSCONTAINER** angegebene Container wurde nicht gefunden.

3

Der durch den Parameter **DATCONTAINER** angegebene Container wurde nicht gefunden.

22 LENGERR

RESP2-Werte:

1

Die Daten in dem durch den Parameter **DATCONTAINER** angegebenen Container sind zu kurz für den angegebenen Umwandlungstyp.

2

Der Puffer für den Wert in **ELEMNAME** ist zu klein.

3

Der Puffer für den Wert in **ELEMNS** ist zu klein.

4

Der Puffer für den Wert in **TYPENAME** ist zu klein.

5

Der Puffer für den Wert in **TYPENS** ist zu klein.

16 INVREQ

RESP2-Werte:

1

Die XMLTRANSFORM-Ressource ist nicht aktiviert.

2

Der mit **XMLCONTAINER** angegebene Container ist leer.

3

Die XML-Eingabedaten sind falsch. Weitere Informationen enthält die Fehlermeldung im Container DFH-XML-ERRORMSG.

4

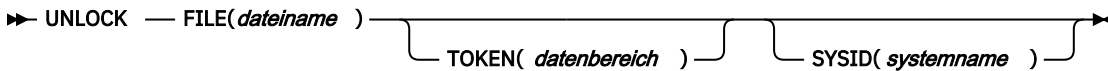
Die Eingabedaten können nicht konvertiert werden. Weitere Informationen enthält die Fehlermeldung im Container DFH-XML-ERRORMSG.

- 5** Die Anwendungsdaten sind falsch. Weitere Informationen enthält die Fehlermeldung im Container DFH-XML-ERRORMSG.
- 6** Die Anwendungsdaten können nicht konvertiert werden. Weitere Informationen enthält die Fehlermeldung im Container DFH-XML-ERRORMSG.
- 7** Entweder der XML-Container oder der für NAMESPACE angegebene Container wurde nicht im Zeichenmodus (CHAR) gefüllt.
- 9** Die XMLTRANSFORM-Ressource unterstützt das angeforderte XML-Element nicht.
- 10** Die XMLTRANSFORM-Ressource unterstützt den angeforderten XML-Typ nicht.
- 11** Beim Herstellen der Verbindung zu einem vom Anbieter bereitgestellten Umsetzungsprogramm trat ein Problem auf.
- 13** Der Parameter **CHANNEL** wurde nicht angegeben und ist erforderlich.
- 14** Der Parameter **ELEMNAME** wurde nicht angegeben und ist erforderlich.
- 15** Der Parameter **ELEMNS** wurde nicht angegeben und ist erforderlich.
- 16** Der Parameter **DATCONTAINER** wurde nicht festgelegt und ist erforderlich.
- 17** Bei der Laufzeitvalidierung trat ein Fehler auf.
- 18** Hinsichtlich des Datentyps für den Container trat ein Fehler auf.
- 101** Der Benutzer ist nicht zur Verwendung der XMLTRANSFORM-Ressource berechtigt.

UNLOCK

Gibt die exklusive Steuerung frei.

UNLOCK



Bedingungen: DISABLED, FILENOTFOUND, ILLOGIC, INVREQ, IOERR, ISCINVREQ, NOTAUTH, NOT-OPEN, SYSIDERR

Dieser Befehl ist threadsicher, wenn für die Datei, auf die er sich bezieht, Folgendes zutrifft:

- Sie ist als ferne Datei definiert und der Befehl wird per Funktionsverlagerung über eine IPIC-Verbindung an eine ferne CICS-Region übertragen.
- Sie ist als lokale VSAM-Datei, als VSAM-RLS-Datei oder als Coupling-Facility-Datentabelle definiert.

Dieser Befehl ist nicht threadsicher, wenn für die Datei, auf die er sich bezieht, Folgendes zutrifft:

- Sie ist als ferne Datei definiert und der Befehl wird per Funktionsverlagerung über eine Verbindung übertragen, die keine IPIC-Verbindung ist.
- Sie ist als gemeinsam genutzte Datentabelle oder als BDAM-Datei definiert.

Beschreibung

Der Befehl **UNLOCK** gibt die exklusive Steuerung frei, die als Reaktion auf einen Lesebefehl mit der Option **UPDATE** eingerichtet wurde. Sie verwenden den Befehl, wenn Sie einen Datensatz zur Aktualisierung aufrufen und anschließend entscheiden, dass der Datensatz doch nicht aktualisiert werden soll. Bei einer wiederherstellbaren Datei (also einer Datei, die keine Coupling-Facility-Datentabelle referenziert) bleibt die Ressource jedoch gesperrt, bis entweder ein Befehl **SYNCPOINT** ausgeführt oder die Task beendet wird. Der Datensatz kann sich in einer Datei bzw. in einer von CICS verwalteten oder einer benutzerverwalteten Datentabelle auf einem lokalen oder einem fernen System befinden.

Falls sich der Befehl **UNLOCK** auf einen Datensatz bezieht, der sich in einer wiederherstellbaren Coupling-Facility-Datentabelle befindet, wird die Datensatzsperre sofort freigegeben, falls die Task in der aktuellen Einheit zuvor keine Änderungen an diesem Datensatz vorgenommen oder ihn als neuen Datensatz hinzugefügt hat. Für den Fall, dass der Datensatz geändert oder neu zur Tabelle hinzugefügt worden ist, bleibt er gesperrt, bis entweder ein Befehl **SYNCPOINT** ausgeführt oder die Task beendet wird.

Falls ein Befehl **UNLOCK** kein Token enthält, wird versucht, ihn entweder mit einem Lesevorgang mit der Option **UPDATE** der ebenfalls kein Token enthält, oder mit einer Operation **WRITE MASSINSERT** abzugleichen. Falls weder ein solcher Lesevorgang noch eine solche Operation gefunden wird, wird keine Aktion ausgeführt und die Antwort **NORMAL** zurückgegeben.

Mit diesem Befehl können Sie eine VSAM-Operation **WRITE MASSINSERT** für eine VSAM-Datei beenden.

Sperren bei Aktualisierung in Suchvorgängen freigeben

Der Befehl **UNLOCK** gibt keine Sperren frei, die für Datensätze infolge von Befehlen **READNEXT** oder **READPREV** eingerichtet wurden, bei denen die Option **UPDATE** angegeben war. In diesem Fall wird lediglich der Wert der Option **TOKEN** ausgewertet, weshalb er nicht zur Ausführung einer Aktualisierung verwendet werden kann.

Optionen

FILE(dateiname)

Gibt den Namen der Datei an, die freigegeben werden soll.

Bei Angabe eines Wertes für **SYSID** wird davon ausgegangen, dass sich die Datei, auf die sich dieser Dateiname bezieht, auf einem fernen System befindet, wobei es ohne Belang ist, ob der Name für CICS definiert ist. Ansonsten wird die Ressourcendefinition verwendet, um zu ermitteln, ob sich die Datei (Dataset) auf einem lokalen oder fernen System befindet.

SYSID(systemname)

Gibt den Namen des Systems an, an das die Anforderung gerichtet ist.

TOKEN(datenbereich)

Gibt in Form eines Vollwort-Binärwertes eine eindeutige Kennung für einen Befehl UNLOCK an, die den Befehl einem vorherigen Befehl READ, READNEXT oder READPREV zuordnet, bei dem die Option UPDATE angegeben war.

Falls Sie den Befehl UNLOCK mit dem Wert für TOKEN angeben, der von einem Befehl READNEXT UPDATE oder READPREV UPDATE für eine Datei zurückgegeben wurden, auf die der Zugriff im RLS-Modus erfolgt, wertet der Befehl UNLOCK lediglich den Wert der Option TOKEN aus, weshalb er nicht zur Ausführung einer Aktualisierung verwendet werden kann. Die Datensatzsperre wird nicht durch den Befehl freigegeben.

TOKEN kann durch Funktionsverlagerung übertragen werden. Falls jedoch eine Anforderung mit Angabe von TOKEN per Funktionsverlagerung an eine CICS-Region übertragen wird, die dieses Schlüsselwort nicht erkennt, schlägt die Anforderung fehl.

Bedingungen**84 DISABLED**

RESP2-Werte:

50

Eine Datei ist inaktiviert, weil sie ursprünglich als inaktiviert definiert wurde und seit dem nicht aktiviert wurde.

Eine Datei wurde durch einen EXEC CICS-Befehl **SET FILE** oder einen Befehl **CEMT SET FILE** inaktiviert.

Diese Bedingung kann nicht auftreten, wenn der Befehl UNLOCK auf einen erfolgreichen Lesebefehl zur Aktualisierung oder auf eine VSAM-Operation WRITE MASSINSERT folgt.

Standardaktion: Die Task wird abnormal beendet.

12 FILENOTFOUND

RESP2-Werte:

1

Ein in der Option FILE referenzierter Dateiname ist nicht für CICS definiert und die Option SYSID wurde nicht angegeben.

Standardaktion: Die Task wird abnormal beendet.

21 ILLOGIC

RESP2-Werte (für VSAM und von CICS verwaltete Datentabellen):

110

Es ist ein VSAM-Fehler aufgetreten, der zu keiner der anderen CICS-Antwortkategorien gehört.

Weitere Informationen sind bei EIBRCODE im EXEC-Schnittstellenblock verfügbar; Details finden Sie unter [EIB-Felder](#).

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

RESP2-Werte:

47

Ein Befehl UNLOCK enthielt ein Token, dessen Wert mit keinem der Tokens übereinstimmt, die bei einem vorhandenen Befehl READ mit der Option UPDATE verwendet werden.

48

Es wurde versucht, eine Anforderung, die ein Schlüsselwort TOKEN enthält, per Funktionsverlagerung zu übertragen.

Standardaktion: Die Task wird abnormal beendet.

17 IOERR

RESP2-Werte:

120

Während der Dateisteuerungsoperation ist ein E/A-Fehler aufgetreten. Ein E/A-Fehler ist ein nicht erwartetes Ereignis, das nicht durch eine CICS-Bedingung abgedeckt ist.

Für VSAM-Dateien weist IOERR in der Regel auf einen Hardwarefehler hin. Weitere Informationen sind im EXEC-Schnittstellenblock verfügbar. Einzelheiten finden Sie unter [EIB-Felder](#).

Bei einer Coupling-Facility-Datentabelle gibt eine Bedingung IOERR an, dass eine falsche Antwort von einem Coupling-Facility-Zugriff zurückgegeben wurde.

Standardaktion: Die Task wird abnormal beendet.

54 ISCINVREQ

RESP2-Werte:

70

Das ferne System gibt einen Fehler an, der keiner bekannten Bedingung entspricht.

Standardaktion: Die Task wird abnormal beendet.

70 NOTAUTH

RESP2-Werte:

101

Eine Ressourcensicherheitsprüfung ist für FILE(dateiname) fehlgeschlagen.

Standardaktion: Die Task wird abnormal beendet.

19 NOTOPEN

RESP2-Werte:

60

Die Bedingung NOTOPEN (RESP2 60) wird aus einem der folgenden Gründe zurückgegeben:

- Die angeforderte Datei ist geschlossen (CLOSED) und nicht aktiviert (UNENABLED). Der Status CLOSED, UNENABLED wird erreicht, nachdem eine Anforderung CLOSE für eine Datei mit dem Status OPEN ENABLED empfangen wurde und die Datei nicht mehr verwendet wird. Sie können CLOSED, UNENABLED auch als Anfangsstatus festlegen, indem Sie in der FILE-Ressourcendefinition STATUS(UNENABLED) und OPENTIME(FIRSTREF) angeben. (Bei BDAM-Dateien verwenden Sie den Parameter FILSTAT des Makros DFHFCT (TYPE=FILE)).
- Die angeforderte Datei besitzt den Status OPEN (geöffnet) und wird durch andere Transaktionen verwendet, aber für die Datei wurde eine Anforderung CLOSE empfangen.
- Ein Befehl UNLOCK wurde für eine Datei ausgegeben, die sich aufgrund eines Befehls SET DSNNAME QUIESCED oder IMMQUIESCED im Quiescemodus befindet oder gerade in diesen Modus versetzt wird.
- Die angeforderte Datei hat den Status CLOSED (geschlossen) und ENABLED (aktiviert), weshalb CICS versucht hat, die Datei im Rahmen der Anforderungsausführung zu öffnen. Das Öffnen der Datei ist aus einem nicht näher bezeichneten Grund fehlgeschlagen. Sie sollten in der Konsole nach Nachrichten suchen, die erläutern, warum die Datei nicht erfolgreich geöffnet werden konnte.

Diese Bedingung tritt nicht auf, wenn die Anforderung für eine geschlossene, nicht aktivierte Datei (CLOSED, DISABLED) abgesetzt wird. In diesem Fall tritt die Bedingung DISABLED auf.

Diese Bedingung kann ebenfalls nicht auftreten, wenn der Befehl UNLOCK auf einen erfolgreichen Befehl READ zur Aktualisierung oder auf eine Operation WRITE MASSINSERT folgt.

Standardaktion: Die Task wird abnormal beendet.

53 SYSIDERR

RESP2-Werte:

130

Die Option SYSID gibt einen Namen an, bei dem es sich weder um die lokale CICS-Region noch um ein fernes System handelt, das für CICS durch eine CONNECTION-Definition definiert ist. Die Bedingung SYSIDERR tritt außerdem auf, wenn die Verbindung zu dem fernen System geschlossen ist.

131

Für eine Coupling-Facility-Datentabelle ist die Verbindung zum Server für Coupling-Facility-Datentabellen fehlgeschlagen. Dies könnte dadurch verursacht worden sein, dass der Server selbst fehlgeschlagen ist, oder dass der Server zwar verfügbar ist, CICS jedoch keine Verbindung zu ihm herstellen konnte.

132

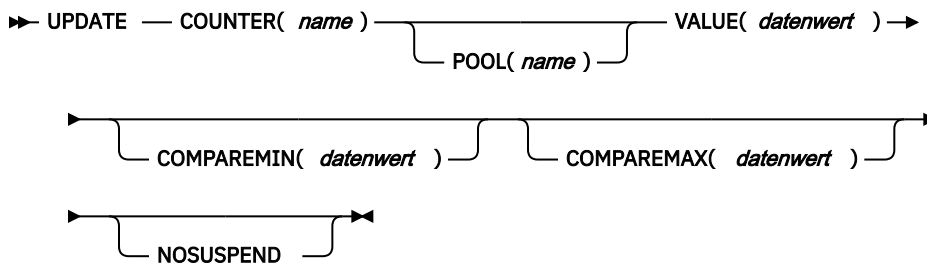
Der Befehl UNLOCK wurde für eine Coupling-Facility-Datentabelle ausgegeben, die nicht mehr vorhanden ist. Vermutlich liegt dies an einem Fehler der Coupling-Facility, durch den der Server für Coupling-Facility-Datentabellen ebenfalls fehlschlägt. Informationen zum erneuten Starten eines Servers für Coupling-Facility-Datentabellen und zum erneuten Laden einer Tabelle finden Sie unter [Serverregion für Coupling-Facility-Datentabellen definieren und starten](#).

Standardaktion: Die Task wird abnormal beendet.

UPDATE COUNTER und UPDATE DOUNTER

Aktualisiert den gegenwärtigen Wert eines benannten Zählers. COUNTER wird für Vollwortzähler mit Vorzeichen verwendet, DOUNTER für Doppelwortzähler ohne Vorzeichen.

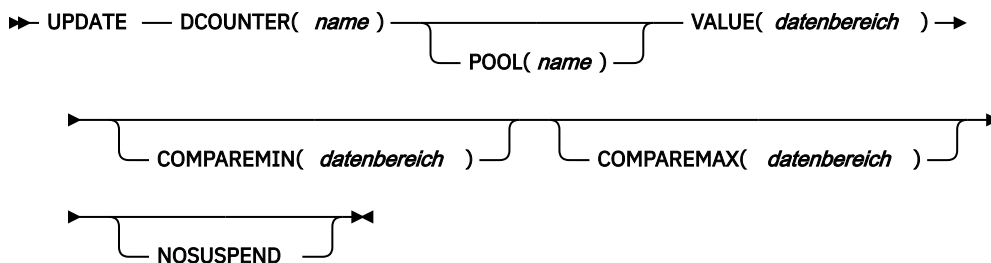
UPDATE COUNTER



Bedingungen: BUSY, INVREQ, SUPPRESSED

Dieser Befehl ist threadsicher.

UPDATE DOUNTER



Bedingungen: BUSY, INVREQ, SUPPRESSED

Dieser Befehl ist threadsicher.

Beschreibung

Dieser Zählerbefehl legt einen neuen aktuellen Wert für den benannten Zähler fest.

Mit den Optionen COMPAREMAX und COMPAREMIN können Sie nur einen neuen aktuellen Wert festlegen, der innerhalb eines bestimmten Bereichs oder über bzw. unter einem angegebenen Wert liegt.

Informationen zur Angabe von Vollwort- und Doppelwortvariablen bei diesen Befehlen für benannte Zähler finden Sie unter „Argumentwerte für CICS-Befehle“ auf Seite 2.

Optionen

COMPAREMAX(*datenwert*)

Gibt als Vollwort-Binärwert mit Vorzeichen (oder bei DCOUNTER als Doppelwort-Binärwert ohne Vorzeichen) einen Wert an, der mit dem aktuellen Wert des benannten Zählers verglichen werden soll und legt das Ergebnis des Befehls UPDATE als Bedingung für den Vergleich fest:

- Falls der zuzuordnende aktuelle Wert kleiner-gleich dem Wert ist, der für den Parameter COMPAREMAX angegeben ist, wird der aktuelle Wert zurückgesetzt und eine normale Antwort ausgegeben.
- Wenn der aktuelle Wert größer als der angegebene Wert ist, gibt CICS eine Ausnahmebedingung zurück.

Normalerweise ist der Wert von COMPAREMAX größer als der Wert von COMPAREMIN und der aktuelle Wert muss beide Vergleichskriterien erfüllen (d. h., er muss zwischen den beiden Werten liegen oder mit einem der Werte identisch sein).

Sie können einen Wert für COMPAREMAX angeben, der kleiner als der Wert für COMPAREMIN ist. In dieser Situation wird für den aktuellen Wert angenommen, dass er im gültigen Bereich liegt, wenn er entweder den mit COMPAREMIN oder den mit COMPAREMAX angegebenen Vergleich erfüllt.

COMPAREMIN(*datenwert*)

Gibt als Vollwort-Binärwert mit Vorzeichen (oder bei DCOUNTER als Doppelwort-Binärwert ohne Vorzeichen) einen Wert an, der mit dem aktuellen Wert des benannten Zählers verglichen werden soll und legt das Ergebnis des Befehls UPDATE als Bedingung für den Vergleich fest:

- Falls der zuzuordnende aktuelle Wert größer-gleich dem Wert ist, der für den Parameter COMPAREMIN angegeben ist, wird der aktuelle Wert zurückgesetzt und eine normale Antwort ausgegeben.
- Wenn der aktuelle Wert kleiner als der angegebene Wert ist, gibt CICS eine Ausnahmebedingung zurück.

Anmerkung: Sie können einen Wert für COMPAREMIN angeben, der größer als der Wert für COMPAREMAX ist. Informationen zur Auswirkung einer solchen Angabe finden Sie unter dem Parameter COMPAREMAX.

COUNTER(*name*)

Gibt den Namen des benannten Zählers an, dessen aktuelle Zahl auf den Wert zurückgesetzt werden soll, der im Parameter VALUE angegeben ist. Der Name kann bis zu 16 alphanumerische Zeichen umfassen. Falls es sich bei *name* um eine Variable handelt, die einen weniger als 16 Zeichen langen Namen enthält, muss der Name mit nachgestellten Leerzeichen aufgefüllt werden.

DCOUNTER(*name*)

Gibt den Namen des benannten Zählers an, dessen aktuelle Zahl auf den Wert zurückgesetzt werden soll, der im Parameter VALUE angegeben ist. Der Name kann bis zu 16 alphanumerische Zeichen umfassen. Falls es sich bei *name* um eine Variable handelt, die einen weniger als 16 Zeichen langen Namen enthält, muss der Name mit nachgestellten Leerzeichen aufgefüllt werden.

NOSUSPEND

Gibt eine sofortige Rückkehr zum Anwendungsprogramm während einer Neuerstellung der Coupling-Facility-Struktur an. Wenn Sie den Parameter NOSUSPEND nicht angeben, wird die Anforderung so lange wiederholt, bis sie erfolgreich ist. Wenn Sie NOSUSPEND angeben, schlägt die Anforderung sofort fehl und CICS gibt die Bedingung BUSY, RESP=500 zurück.

POOL(*poolname*)

Gibt den Namen des Pools an, in dem sich der benannte Zähler befindet.

Gültige Zeichen für die Poolselektorzichenfolge sind A bis Z, 0 bis 9, \$ @ # und _ (Unterstreichungszeichen). Falls es sich bei *name* um eine Variable handelt, die einen weniger als 8 Zeichen langen Namen enthält, muss der Name mit nachgestellten Leerzeichen aufgefüllt werden.

Dieser Parameter ist optional. Wenn Sie den Namen des Pools nicht angeben, wird ein Poolauswahlwert aus 8 Leerzeichen angenommen.

Wenn kein entsprechender Eintrag in der Optionstabelle DFHNCOPT vorhanden ist, verwendet CICS den Namen, der im Systeminitialisierungsparameter NCPLDFT angegeben ist, der den Standardpool für benannte Zähler angibt.

Informationen zum Generieren einer Optionstabelle für benannte Zähler mithilfe des Makros DFHNCO finden Sie unter [Server für benannte Zähler](#).

VALUE(datenwert)

Gibt als Vollwort-Binärwert mit Vorzeichen (beim Befehl COUNTER) bzw. als Doppelwortwert ohne Vorzeichen (beim Befehl DCOUNTER) die neue Zahl an, die als aktueller Wert für den benannten Zähler festgelegt werden soll.

Bedingungen

128 BUSY

RESP2-Werte:

500

Die Option NOSUSPEND wurde im Befehl angegeben und die Coupling-Facility-Struktur ist momentan während einer Neuerstellung nicht verfügbar.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

RESP2-Werte:

201

Der benannte Zähler wurde nicht gefunden.

301

Der Server hat einen Fehlercode gemeldet, der für die Schnittstelle des benannten Zählers unverständlich ist. Dies ist in der Regel nur dann möglich, wenn das Schnittstellenlademodul DFHNCIF eine geringere Wartungsstufe oder ein niedrigeres Release-Level als der Server selbst aufweist.

303

Bei einem Makro, das für den Zugriff auf die Coupling-Facility verwendet wird, trat ein unerwarteter Fehler wie beispielsweise eine Strukturstörung oder eine Verbindungsunterbrechung auf. Weitere Informationen enthält die Nachricht DFHNC0441 im Jobprotokoll der Anwendung.

304

Der im Programm angegebene Parameter für die Poolauswahl kann anhand der aktuellen Optionstabelle nicht in einen gültigen Servernamen aufgelöst werden.

305

Die Schnittstelle kann keine Verbindung zum Server für den ausgewählten Pool mit benannten Zählern herstellen. Weitere Informationen enthält eine AXM-Servicenachricht (AXMSCnnnn) im Jobprotokoll der Anwendung.

306

Während der Server eine Anforderung verarbeitete, trat eine abnormale Beendigung auf. Weitere Informationen enthält eine Nachricht im Jobprotokoll der Anwendung und im Jobprotokoll des Servers.

308

Das Optionstabellenmodul DFHNCOPT, das zum Auflösen eines Poolnamens benötigt wird, konnte nicht geladen werden.

309

Während der Verarbeitung der Optionstabelle hat die Schnittstelle für benannte Zähler ein unbekanntes Eintragsformat festgestellt. Entweder wurde die Optionstabelle nicht ordnungsgemäß generiert oder das Schnittstellenlademodul DFHNCIF hat nicht dasselbe Release-Level wie die Optionstabelle.

310

Ein Optionstabelleneintrag, der mit dem angegebenen Poolnamen übereinstimmt, hat ein Benutzerexitprogramm angegeben, aber das Benutzerexitprogramm ist nicht durch eine Linkbearbeitung mit der Optionstabelle verbunden und kann nicht geladen werden.

311

Eine Antwort vom Server des benannten Zählers an das Schnittstellenmodul DFHNCIF der Clientregion gibt an, dass gegenwärtig ein systemverwalteter erneuter Build ausgeführt wird, aber die Bedingung wird von der EXEC CICS-Schnittstelle nicht erkannt. Dies bedeutet, dass die CICS-Region den Stand von CICS TS 2.1 oder früher aufweist.

403

Der Parameter POOL enthält ungültige Zeichen oder eingebettete Leerzeichen.

404

Der Parameter COUNTER enthält ungültige Zeichen oder eingebettete Leerzeichen.

406

Der Parameter VALUE ist ungültig. Sie können den aktuellen Wert nicht kleiner als den Mindestwert oder größer als den Maximalwert plus 1 festlegen.

Standardaktion: Die Task wird abnormal beendet.

72 SUPPRESSED

RESP2-Werte:

103

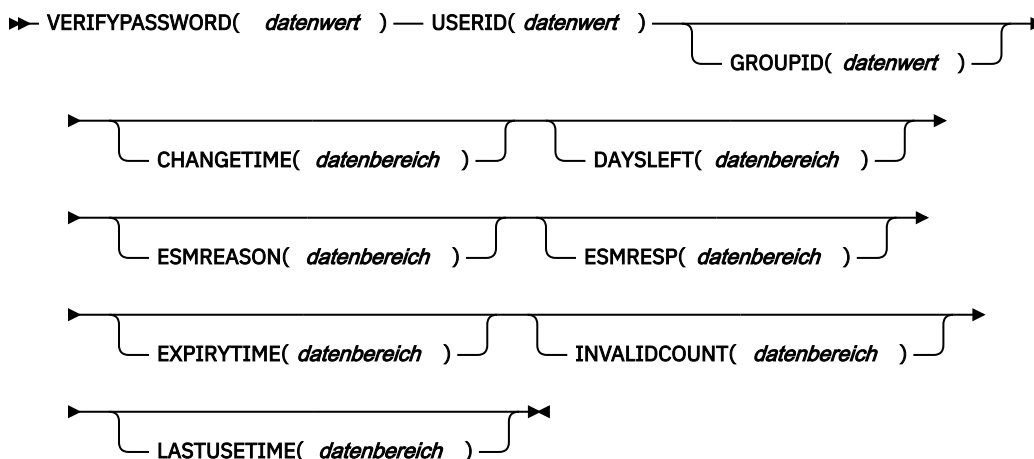
Diese Bedingung tritt in einer der folgenden Situationen auf:

- Der aktuelle Wert des benannten Zählers liegt nicht in dem durch die Parameter COMPAREMAX und COMPAREMIN angegebenen Bereich, falls beide Parameter angegeben sind.
- Der aktuelle Wert des benannten Zählers ist größer als der Wert des Parameters COMPAREMAX oder kleiner als der Wert des Parameters COMPAREMIN, falls nur eine der beiden Optionen angegeben ist.

Standardaktion: Die Task wird abnormal beendet.

VERIFY PASSWORD

Überprüft, ob ein Kennwort mit dem Kennwort übereinstimmt, das durch einen *externen Sicherheitsmanager (ESM)* aufgezeichnet wurde.

VERIFY PASSWORD

Bedingungen: INVREQ, NOTAUTH, USERIDERR

Dieser Befehl ist threadsicher.

Beschreibung

Mit dem Befehl **VERIFY PASSWORD** können Sie überprüfen, ob ein Kennwort mit demjenigen Kennwort übereinstimmt, das durch einen externen Sicherheitsmanager für eine Benutzer-ID aufgezeichnet wurde. Der Befehl gibt die Werte zurück, die durch den externen Sicherheitsmanager für das Kennwort aufgezeichnet wurden. Dieser Prozess wird als 'Kennwortüberprüfung' bezeichnet. Falls Ihr System Kennwortphrasen zusätzlich zu oder anstelle von Standardkennwörtern einsetzt, verwenden Sie statt des Befehls **VERIFY PASSWORD** den Befehl **VERIFY PHRASE**.



Achtung: Um sicherzustellen, dass Kennwörter nicht in System- oder Transaktionsspeicherausgängen offengelegt werden, löschen Sie den Inhalt von Kennwort- bzw. Kennwortphrasenfeldern in den EXEC CICS-Befehlen, die eine Option für Kennwort oder Kennwortphrase haben, möglichst bald nach der Verwendung.

Ziehen Sie aus einer erfolgreichen Anforderung **VERIFY PASSWORD** nicht den Schluss, dass die Benutzer-ID auch zur Anmeldung bei der CICS-Region mit dem EXEC CICS-Befehl **SIGNON** verwendet werden könnte. Eine Kennwortüberprüfungsanforderung kann beispielsweise die folgenden Probleme nicht feststellen:

- Die Verbindungen der Benutzer-ID zu Gruppen wurden entzogen.
- Die Benutzer-ID ist nicht für den Zugriff auf den CICS-Adressraum (angegeben durch die Anwendungs-ID) berechtigt.
- Die Benutzer-ID ist nicht zur Verwendung des Terminals berechtigt, an dem sich der Benutzer anmeldet (dies wird durch die Klasse TERMINAL angegeben).

Falls Sie in der Anforderung **VERIFY PASSWORD** ein falsches Kennwort angeben, wird die Anzahl der ungültigen Versuche für die Benutzer-ID erhöht. Wenn Sie bei aufeinanderfolgenden Anforderungen **VERIFY PASSWORD** mehrere falsche Kennwörter angeben, wird die Benutzer-ID möglicherweise durch den externen Sicherheitsmanager entzogen. CICS gibt die Nachricht DFHXS1201 aus, wenn Sie bei einer Anforderung **VERIFY PASSWORD** ein falsches Kennwort angeben. Wenn Sie nach einem oder mehreren ungültigen Versuchen ein richtiges Kennwort angeben, gibt CICS die Nachricht DFHXS1206 aus, die eine Anzahl der ungültigen Versuche enthält.

Anders als der EXEC CICS-Befehl **SIGNON** ist der Befehl **VERIFY PASSWORD** nicht von der Hauptfunktion abhängig und kann daher auch in Umgebungen ausgegeben werden, bei denen es sich nicht um Terminals handelt (z. B. in Webanwendungen).

CICS erzwingt an jedem Tag eine Anforderung für eine vollständige Überprüfung, wenn eine Benutzer-ID zum ersten Mal für die Anmeldung bei der CICS-Region verwendet oder mit einem Befehl **VERIFY PASSWORD** überprüft wird. Die Anforderung für die vollständige Überprüfung zeichnet das Datum und die Uhrzeit des letzten Zugriffs für die Benutzer-ID auf und schreibt Benutzerstatistikdaten. Eine vollständige Überprüfung findet ebenfalls statt, wenn ein falsches Kennwort eingegeben wurde und anschließend eine erfolgreiche Anforderung erfolgt. In anderen Fällen verwendet der Befehl **VERIFY PASSWORD** für die Überprüfung des Kennworts ein Schnellverfahren. Details zu den verwendeten SAF-Schnittstellen enthält der Abschnitt [Steuerpunkte für CICS-Sicherheit](#).

Die Option GROUPID gibt eine Gruppe an, die bei der Überprüfung des Kennworts einbezogen werden soll. Wenn die Anforderung **VERIFY PASSWORD** bei Verwendung der Option GROUPID erfolgreich ist, kann daraus nicht geschlossen werden, dass die Gruppen-ID ebenfalls zur Anmeldung bei der CICS-Region mit dem EXEC CICS-Befehl **SIGNON** verwendet werden kann.

Anmerkung: In den Optionen CHANGETIME, LASTUSETIME und EXPIRYTIME hat der zurückgegebene Zeitwert dasselbe Format wie beim Befehl **ASKTIME**, wird also in Einheiten des Typs ABSTIME angegeben. Weitere Informationen zum Format ABSTIME finden Sie unter [ASKTIME](#). Mithilfe des Befehls **FORMATTIME** können die Daten in einem bestimmten Datums-/Zeitformat neu formatiert werden.

Optionen

CHANGETIME(datenbereich)

Gibt das Datum und die Uhrzeit für die letzte Änderung des Kennworts in Einheiten des Typs ABSTIME zurück.

Wenn RACF als externer Sicherheitsmanager verwendet wird, wird als Zeitpunkt Mitternacht angegeben.

DAYSLEFT(datenbereich)

Gibt als Halbwort-Binärzahlfeld die Anzahl der Tage ab dem aktuellen Datum bis zum Ablauf des Kennworts zurück. Falls das Kennwort nicht abläuft, wird der Wert '-1' zurückgegeben.

ESMREASON(datenbereich)

Gibt in einem Vollwort-Binärfeld den Ursachencode zurück, den CICS vom externen Sicherheitsmanager empfängt.

Falls RACF als externer Sicherheitsmanager verwendet wird, enthält dieses Feld den RACF-Ursachencode.

Der externe Sicherheitsmanager gibt nicht immer Antwort- und Ursachencodes an CICS zurück. Überprüfen Sie neben den Werten von ESMRESP und ESMREASON unbedingt auch die Werte von EIBRESP und EIBRESP2, die durch diesen Befehl zurückgegeben werden.

ESMRESP(datenbereich)

Gibt in einem Vollwort-Binärfeld den Antwortcode zurück, den CICS vom externen Sicherheitsmanager empfängt.

Falls RACF als externer Sicherheitsmanager verwendet wird, enthält dieses Feld den RACF-Antwortcode.

Der externe Sicherheitsmanager gibt nicht immer Antwort- und Ursachencodes an CICS zurück. Überprüfen Sie neben den Werten von ESMRESP und ESMREASON unbedingt auch die Werte von EIBRESP und EIBRESP2, die durch diesen Befehl zurückgegeben werden.

EXPIRYTIME(datenbereich)

Gibt das Datum und die Uhrzeit für den Ablauf des Kennworts in Einheiten des Typs ABSTIME zurück.

Wenn RACF als externer Sicherheitsmanager verwendet wird, wird als Zeitpunkt Mitternacht angegeben.

GROUPID(datenwert)

Gibt die achtstellige Gruppen-ID des Benutzers an, dessen Kennwort überprüft werden soll.

INVALIDCOUNT(datenbereich)

Gibt als Halbwort-Binärzahlfeld an, wie oft für diesen Benutzer ein ungültiges Kennwort eingegeben wurde.

LASTUSETIME(datenbereich)

Gibt das Datum und die Uhrzeit für den letzten Zugriff auf diese Benutzer-ID in Einheiten des Typs ABSTIME zurück.

PASSWORD(datenwert)

Gibt das achtstellige Kennwort an, das der externe Sicherheitsmanager für die angegebene Benutzer-ID überprüfen soll. Falls das Kennwort nicht gültig ist, werden die übrigen Daten nicht zurückgegeben.

Falls der externe Sicherheitsmanager Kennwörter in Groß-/Kleinschreibung nicht zulässt, wird das Kennwort in Großbuchstaben konvertiert.

USERID(datenwert)

Gibt die achtstellige Benutzer-ID des Benutzers an, dessen Kennwort überprüft werden soll.

Die angegebene Benutzer-ID wird in Großbuchstaben konvertiert.

Bedingungen**16 INVREQ**

RESP2-Werte:

13

Der vom externen Sicherheitsmanager zurückgegebene Wert in der Option ESMRESP ist nicht von CICS klassifiziert. Eine Erläuterung der Optionen ESMRESP und ESMREASON finden Sie in der ESM-Dokumentation.

18

Die CICS ESM-Schnittstelle ist nicht initialisiert.

29

Der externe Sicherheitsmanager antwortet nicht.

32

Das Benutzer-ID-Feld enthält an einer ungültigen Position ein Leerzeichen.

Standardaktion: Die Task wird abnormal beendet.

70 NOTAUTH

RESP2-Werte:

2

Das angegebene Kennwort ist falsch. Wenn RACF als externer Sicherheitsmanager verwendet wird, wird der von RACF verwaltete Zähler für den Entzug des Kennworts inkrementiert.

Lautet der RESP-Wert des externen Sicherheitsmanagers jedoch 24, wird die Anzahl für den Entzug des Kennworts nicht erhöht.

3

Ein neues Kennwort ist erforderlich.

19

Die Benutzer-ID wurde entzogen.

20

Die Verbindung des Benutzers mit der Standardgruppe wurde entzogen.

23

Der Wert von GROUPID ist in Verbindung mit den Werten von USERID und PASSWORD ungültig.

RESP-Werte des externen Sicherheitsmanagers:

24

Die Anforderung RACROUTE REQUEST=VERIFY(X) ist aufgrund einer Installationsexitroutine ICH-RIX01 fehlgeschlagen.

Standardaktion: Die Task wird abnormal beendet.

69 USERIDERR

RESP2-Werte:

8

Die Benutzer-ID ist dem externen Sicherheitsmanager nicht bekannt.

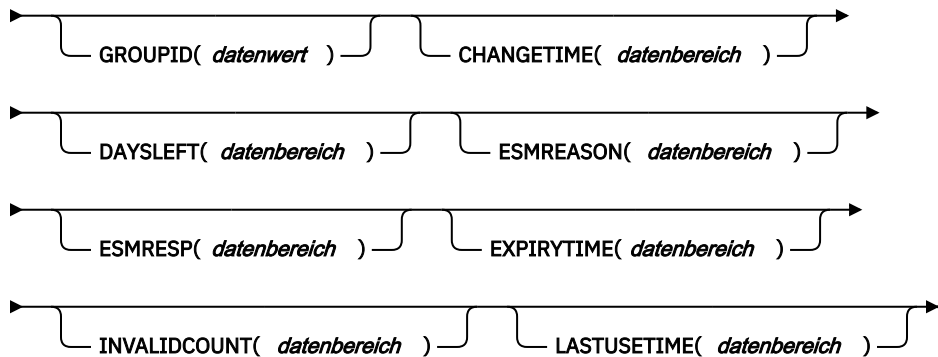
Standardaktion: Die Task wird abnormal beendet.

VERIFY PHRASE

Überprüft, ob ein Kennwort oder eine Kennwortphrase mit dem Kennwort bzw. der Kennwortphrase übereinstimmt, das/die in einem *externen Sicherheitsmanager (ESM)* aufgezeichnet wurde.

VERIFY PHRASE

➔ VERIFYPHRASE(*datenbereich*) — PHRASELEN(*datenwert*) — USERID(*datenwert*) ➔



Bedingungen: INVREQ, LENGERR, NOTAUTH, USERIDERR

Dieser Befehl ist threadsicher.

Beschreibung

Mit dem Befehl **VERIFY PHRASE** können Sie überprüfen, ob ein Kennwort oder eine Kennwortphrase mit demjenigen Kennwort bzw. derjenigen Kennwortphrase übereinstimmt, das/die durch einen externen Sicherheitsmanager für eine Benutzer-ID aufgezeichnet wurde. Der Befehl gibt die Werte zurück, die durch den externen Sicherheitsmanager für das Kennwort bzw. die Kennwortphrase aufgezeichnet wurden. Dieser Prozess wird als 'Kennwortüberprüfung' bezeichnet.



Achtung: Um sicherzustellen, dass Kennwörter nicht in System- oder Transaktionsspeicherausfällen offengelegt werden, löschen Sie den Inhalt von Kennwort- bzw. Kennwortphrasenfeldern in den EXEC CICS-Befehlen, die eine Option für Kennwort oder Kennwortphrase haben, möglichst bald nach der Verwendung.

Eine Benutzer-ID kann sowohl ein Standardkennwort als auch eine Standardkennwortphrase besitzen. Falls die mit der Option PHRASELEN angegebene Länge der Phrase zwischen 1 und 8 Zeichen beträgt, wird die Phrase als Standardkennwort betrachtet und der externe Sicherheitsmanager überprüft, ob der Wert der Option PHRASE mit dem Kennwort übereinstimmt, das vom externen Sicherheitsmanager für die Benutzer-ID aufgezeichnet wurde. Liegt die Länge zwischen 9 und 100 Zeichen, wird der Wert als Kennwortphrase behandelt und der externe Sicherheitsmanager überprüft, ob der Wert von PHRASE mit der Kennwortphrase übereinstimmt, die für die Benutzer-ID aufgezeichnet wurde.

Obwohl das Ablaufintervall für Kennwörter und Kennwortphrasen identisch ist, gibt es separate Werte für die Optionen CHANGETIME und DAYSLEFT, weil sie unabhängig voneinander geändert werden. Die für diese beiden Parameter zurückgegebenen Werte sind davon abhängig, ob im Befehl **VERIFY PHRASE** ein gültiges Kennwort oder eine gültige Kennwortphrase verwendet wird.

Ziehen Sie aus einer erfolgreichen Anforderung **VERIFY PHRASE** nicht den Schluss, dass die Benutzer-ID auch zur Anmeldung bei der CICS-Region mit dem EXEC CICS-Befehl **SIGNON** verwendet werden könnte. Eine Kennwortüberprüfungsanforderung kann beispielsweise die folgenden Probleme nicht feststellen:

- Die Verbindungen der Benutzer-ID zu Gruppen wurden entzogen.
- Die Benutzer-ID ist nicht für den Zugriff auf den CICS-Adressraum (angegeben durch die Anwendungs-ID) berechtigt.
- Die Benutzer-ID ist nicht zur Verwendung des Terminals berechtigt, an dem sich der Benutzer anmeldet (dies wird durch die Klasse TERMINAL angegeben).

Anders als der EXEC CICS-Befehl **SIGNON** ist der Befehl **VERIFY PHRASE** nicht von der Hauptfunktion abhängig und kann daher auch in Umgebungen ausgegeben werden, bei denen es sich nicht um Terminals handelt (z. B. in Webanwendungen).

CICS erzwingt an jedem Tag eine Anforderung für eine vollständige Überprüfung, wenn eine Benutzer-ID zum ersten Mal für die Anmeldung bei der CICS-Region verwendet oder mit einem Befehl **VERIFY PASSWORD** überprüft wird. Die Anforderung für die vollständige Überprüfung zeichnet das Datum und die Uhrzeit des letzten Zugriffs für die Benutzer-ID auf und schreibt Benutzerstatistikdaten. Eine vollständige Überprüfung findet ebenfalls statt, wenn ein falsches Kennwort eingegeben wurde und anschließend eine erfolgreiche Anforderung erfolgt. In anderen Fällen verwendet der Befehl **VERIFY PASSWORD** für die Überprüfung des Kennworts ein Schnellverfahren. Details zu den verwendeten SAF-Schnittstellen enthält der Abschnitt Steuerpunkte für CICS-Sicherheit.

Die Option **GROUPID** gibt eine Gruppe an, die bei der Überprüfung des Kennworts oder der Kennwortphrase einbezogen werden soll. Wenn die Anforderung **VERIFY PHRASE** bei Verwendung der Option **GROUPID** erfolgreich ist, kann daraus nicht geschlussfolgert werden, dass die Gruppen-ID ebenfalls zur Anmeldung bei der CICS-Region mit dem EXEC CICS-Befehl **SIGNON** verwendet werden kann.

Anmerkung: In den Optionen **CHANGETIME**, **LASTUSETIME** und **EXPIRYTIME** hat der zurückgegebene Zeitwert dasselbe Format wie beim Befehl **ASKTIME**, wird also in Einheiten des Typs **ABSTIME** angegeben. Weitere Informationen zum Format **ABSTIME** finden Sie unter **ASKTIME**. Mithilfe des Befehls **FORMATTIME** können die Daten in einem bestimmten Datums-/Zeitformat neu formatiert werden.

Optionen

CHANGETIME(datenbereich)

Gibt das Datum und die Uhrzeit für die letzte Änderung des Kennworts oder der Kennwortphrase in Einheiten des Typs **ABSTIME** zurück.

Wenn RACF als externer Sicherheitsmanager verwendet wird, wird als Zeitpunkt Mitternacht angegeben.

DAYSLEFT(datenbereich)

Gibt als Halbwort-Binärzahlfeld die Anzahl der Tage ab dem aktuellen Datum bis zum Ablauf des Kennworts oder der Kennwortphrase zurück. Falls das Kennwort oder die Kennwortphrase nicht abläuft, wird der Wert '-1' zurückgegeben. Falls ein Benutzer ein Kennwort oder eine Kennwortphrase besitzt, das/die nicht abläuft, ist die Option **DAYSLEFT** ohne Bedeutung und wird mit dem Wert '-1' angegeben.

ESMREASON(datenbereich)

Gibt in einem Vollwort-Binärfeld den Ursachencode zurück, den CICS vom externen Sicherheitsmanager empfängt.

Falls RACF als externer Sicherheitsmanager verwendet wird, enthält dieses Feld den RACF-Ursachencode.

Der externe Sicherheitsmanager gibt nicht immer Antwort- und Ursachencodes an CICS zurück. Überprüfen Sie neben den Werten von **ESMRESP** und **ESMREASON** unbedingt auch die Werte von **EIBRESP** und **EIBRESP2**, die durch diesen Befehl zurückgegeben werden.

ESMRESP(datenbereich)

Gibt in einem Vollwort-Binärfeld den Antwortcode zurück, den CICS vom externen Sicherheitsmanager empfängt.

Falls RACF als externer Sicherheitsmanager verwendet wird, enthält dieses Feld den RACF-Antwortcode.

Der externe Sicherheitsmanager gibt nicht immer Antwort- und Ursachencodes an CICS zurück. Überprüfen Sie neben den Werten von **ESMRESP** und **ESMREASON** unbedingt auch die Werte von **EIBRESP** und **EIBRESP2**, die durch diesen Befehl zurückgegeben werden.

EXPIRYTIME(datenbereich)

Gibt das Datum und die Uhrzeit für den Ablauf des Kennworts in Einheiten des Typs **ABSTIME** zurück.

Wenn RACF als externer Sicherheitsmanager verwendet wird, wird als Zeitpunkt Mitternacht angegeben. Falls ein Benutzer ein Kennwort oder eine Kennwortphrase besitzt, das/die nicht abläuft, ist die Option **EXPIRYTIME** ohne Bedeutung und wird mit dem Wert '-1' angegeben.

GROUPID(datenwert)

Gibt die achtstellige Gruppen-ID des Benutzers an, dessen Kennwortphrase überprüft werden soll.

INVALIDCOUNT(datenbereich)

Gibt als Halbwort-Binärfeld an, wie oft für diesen Benutzer ein ungültiges Kennwort oder eine ungültige Kennwortphrase eingegeben wurde.

LASTUSETIME(datenbereich)

Gibt das Datum und die Uhrzeit für den letzten Zugriff auf diese Benutzer-ID in Einheiten des Typs ABSTIME zurück.

PHRASE(datenbereich)

Gibt ein 1 bis 8 Zeichen langes Kennwort oder eine 9 bis 100 Zeichen lange Kennwortphrase an, das/die durch den externen Sicherheitsmanager erforderlich gemacht wird. Falls die Phrase nicht gültig ist, werden die übrigen Daten nicht zurückgegeben.

Falls der externe Sicherheitsmanager Kennwörter in Groß-/Kleinschreibung nicht zulässt, wird das 1 bis 8 Zeichen lange Kennwort in Großbuchstaben konvertiert.

PHRASELEN(datenwert)

Gibt die Länge des Kennworts bzw. der Kennwortphrase in Form eines Vollwort-Binärwertes an.

USERID(datenwert)

Gibt die Benutzer-ID an, die dem Kennwort oder der Kennwortphrase zugeordnet ist, dessen/deren Überprüfung angefordert wird.

Die angegebene Benutzer-ID wird in Großbuchstaben konvertiert.

Bedingungen**16 INVREQ**

RESP2-Werte:

13

Der vom externen Sicherheitsmanager zurückgegebene Wert in der Option ESMRESP ist nicht von CICS klassifiziert. Eine Erläuterung der Optionen ESMRESP und ESMREASON finden Sie in der ESM-Dokumentation.

18

Die CICS ESM-Schnittstelle ist nicht initialisiert.

29

Der externe Sicherheitsmanager antwortet nicht.

32

Das Feld USERID enthält an einer ungültigen Position ein Leerzeichen.

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

RESP2-Werte:

1

Der Wert für PHRASELEN lag außerhalb des gültigen Bereichs.

70 NOTAUTH

RESP2-Werte:

2

Das angegebene Kennwort bzw. die angegebene Kennwortphrase ist falsch. Wenn RACF als externer Sicherheitsmanager verwendet wird, wird der von RACF verwaltete Zähler für den Entzug des Kennworts inkrementiert.

Lautet der RESP-Wert des externen Sicherheitsmanagers jedoch 24, wird die Anzahl für den Entzug des Kennworts nicht erhöht.

3

Ein neues Kennwort bzw. eine neue Kennwortphrase ist erforderlich.

19

Die Benutzer-ID wurde entzogen.

20

Die Verbindung des Benutzers mit der Standardgruppe wurde entzogen.

23

Der Wert von GROUPID ist in Verbindung mit den Werten von USERID und PHRASE ungültig.

RESP-Werte des externen Sicherheitsmanagers:

24

Die Anforderung RACROUTE REQUEST=VERIFY(X) ist aufgrund einer Installationsexitroutine ICH-RIX01 fehlgeschlagen.

Standardaktion: Die Task wird abnormal beendet.

69 USERIDERR

RESP2-Werte:

8

Die Benutzer-ID ist dem externen Sicherheitsmanager nicht bekannt.

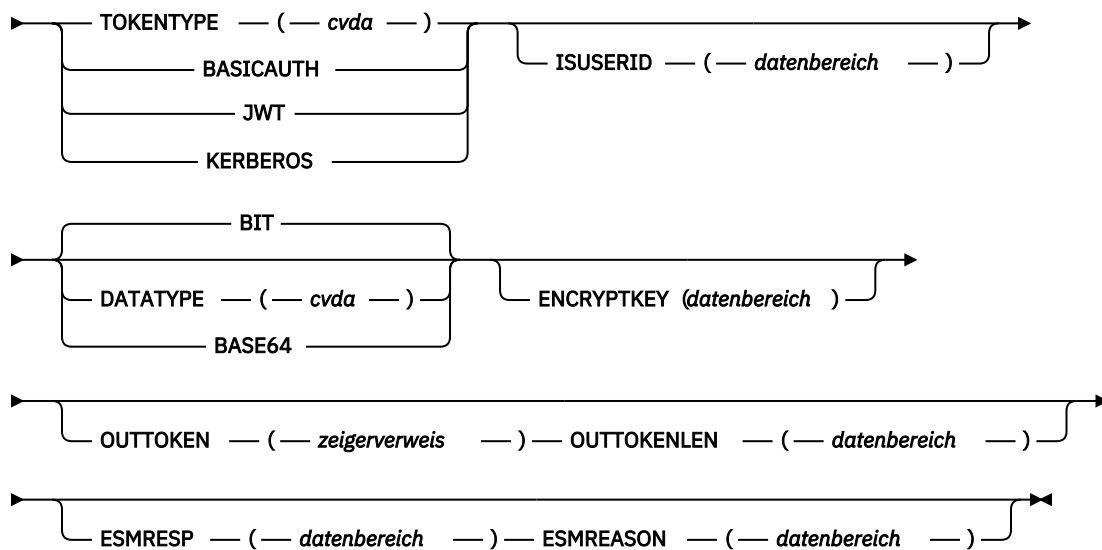
Standardaktion: Die Task wird abnormal beendet.

VERIFY TOKEN

Überprüft, ob ein Sicherheitstoken gültig ist, entsprechend der Bestimmung durch einen externen Sicherheitsmanager (ESM), und ermöglicht dem Aufrufenden, optional Sicherheitsinformationen aus diesem Token zu extrahieren oder zu generieren.

VERIFY TOKEN

➔ VERIFY TOKEN(*datenbereich*) — TOKENLEN(*datenwert*) ➔



Bedingungen: INVREQ, LENGERR, NOTAUTH

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl **VERIFY TOKEN** unterstützt die folgenden Typen von Sicherheitstokens:

- Basisauthentifizierungstokens (BASICAUTH)
- JSON Web Tokens (JWT)
- Kerberos-Tokens (KERBEROS)

Mit dem Befehl **VERIFY TOKEN** wird überprüft, ob ein Token gemäß der Ermittlung durch einen externen Sicherheitsmanager gültig ist. Optional können auch sicherheitsrelevante Informationen ausgegeben werden. Der Befehl ist nicht von der Hauptfunktion abhängig und kann daher auch in Nicht-Terminal-Umgebungen ausgegeben werden, z. B. für die Authentifizierung von Webanforderungen oder Web-Service-Anforderungen. Zusätzliche Angaben über einen Sicherheitsfehler bei diesem Befehl können Sie den Fehlermeldungen entnehmen, die an das CICS-Ziel geschrieben werden.

Unterstützung für Basisauthentifizierungstokens (BASICAUTH)

Ein Basisauthentifizierungstoken ist ein Token mit dem Format *benutzer-id:kennwort*. Das Kennwort kann ein Kennwort, eine Kennphrase, ein PassTicket oder ein MFA-Token sein. Wenn das Token über den Web-Header übertragen wird, kann es der Einfachheit halber im Base 64-Format übergeben werden. Der Hauptzweck der Verwendung des Befehls **VERIFY TOKEN** mit einem Basisauthentifizierungstoken ist es, ein JWT in OUTTOKEN zu generieren. Auf diese Weise können einmalig verwendbare Tokens wie z. B. PassTicket und MFA-Tokens ein kryptographisch sicheres, zeitlich begrenztes Token für die mehrmalige Verwendung generieren.

Unterstützung für JSON Web Tokens (JWT)

Der Befehl **VERIFY TOKEN** überprüft, ob ein JWT-Token gemäß der Ermittlung durch den externen Sicherheitsmanager gültig ist. Er kann darüber hinaus ein JWT generieren. Dies wird unter „[Unterstützung für Basisauthentifizierungstokens \(BASICAUTH\)](#)“ auf Seite 626 beschrieben.

CICS unterstützt nur signierte JWTs. Ein JWT muss das unter [JSON Web Token \(JWT\) Specification RFC 7519](#) beschriebene Format aufweisen. Wenn der externe Sicherheitsmanager RACF ist, müssen Sie Profile in der Klasse IDTDATA anhand der Beschreibung unter [Unterstützung für JWT mit RACF](#) erstellen, damit CICS JWTs unterstützt.

Unterstützung für Kerberos-Tokens (KERBEROS)

Der Befehl **VERIFY TOKEN** überprüft, ob ein Kerberos-Token gemäß der Ermittlung durch einen externen Sicherheitsmanager gültig ist. Optional gibt der Befehl die Benutzer-ID eines Kerberos-Prinzips zurück, die dem Token zugeordnet ist. Falls das Kerberos-Token angibt, dass die gegenseitige Authentifizierung genutzt wird, gibt der Befehl ein Kerberos-Ausgabetoken zurück.

Falls RACF als externer Sicherheitsmanager verwendet wird, muss die CICS-Region, in der der Befehl ausgeführt wird, durch RACF autorisiert sein, damit der Kerberos-Prinzipal für das Token angefordert werden kann. Weitere Informationen finden Sie unter [RACF für Kerberos konfigurieren](#).

Falls die Option **ENCRYPTKEY** angegeben ist, muss der Wert von **TOKEN** ein Kerberos-Token sein, das von einem Sicherheitstokenservice abgerufen wurde, der den Nachrichtenschutz unterstützt.

Falls der Wert von **TOKEN** ein Kerberos-Token ist, das die Nutzung der gegenseitigen Authentifizierung angibt, müssen die Optionen **OUTTOKEN** und **OUTTOKENLEN** angegeben werden. In diesem Fall gibt **OUTTOKEN** ein Ausgabetoken zurück, das von einem externen Sicherheitsmanager wie z. B. z/OS Security Server abgerufen wird, und **OUTTOKENLEN** gibt die Länge des Ausgabetokens zurück. Das aufrufende Programm sollte das Ausgabetoken an das System zurückgeben, von dem das Kerberos-Token bereitgestellt wurde, um die Authentifizierung der CICS-Region zu ermöglichen.

Falls der Wert von **TOKEN** ein Kerberos-Token ist, das nicht die Nutzung der gegenseitigen Authentifizierung angibt, und die Optionen **OUTTOKEN** und **OUTTOKENLEN** angegeben sind, werden in diesen Optionen Nullwerte zurückgegeben.

Optionen

DATATYPE

Gibt den Typ der Daten in der Option TOKEN an. Die CVDA-Werte lauten wie folgt:

BIT

Das Token enthält Bitdaten. Dies ist der Standardwert.

BASE64

Das Token enthält Zeichendaten in Base64-Codierung. Zulässige Zeichen sind A-Z a-z 0-9 + / =.

Falls Ihre Zeichendaten nicht in einer CCSID enthalten sind, die mit US EBCDIC kompatibel ist, müssen Sie sie konvertieren. Zur Konvertierung können Sie die API CONTAINER verwenden.

BASE64 wird nur für die TOKENTYPE-Werte BASICAUTH und KERBEROS unterstützt.

ENCRYPTKEY(datenbereich)

Dies ist das 4 Byte umfassende Token, das zum Verschlüsseln eines zurückgegebenen Passtickets verwendet wird.

ESMREASON(datenbereich)

Gibt in einem Vollwort-Binärfeld den Ursachencode zurück, den CICS vom externen Sicherheitsmanager empfängt.

Falls RACF als externer Sicherheitsmanager verwendet wird, enthält dieses Feld den RACF-Ursachencode.

- Für BASICAUTH und JWT sind die Antwort- und Ursachencodes im Thema zum Befehl [RACROUTE VERIFY](#) im Knowledge Center für z/OS Security Server RACF dokumentiert.
- Für KERBEROS sind die Antwort- und Ursachencodes unter **z/OS Security Server RACF Callable Services > R_GenSec function code 2 GSS-API > Accept a GSS-API security context** dokumentiert. Weitere Informationen finden Sie unter GSS-API-Unterfunktionscodes in [z/OS Security Server RACF Callable Services](#). Erläuterungen für Ursachencodes enthalten die Abschnitte [R_GenSec-Rückgabecodes](#) und [-Ursachencodes](#) in [z/OS Security Server RACF Callable Services](#) und [Statuscodes](#) in [z/OS Integrated Security Services Network Authentication Service Administration](#).

ESMRESP(datenbereich)

Gibt in einem Vollwort-Binärfeld den Antwortcode zurück, den CICS vom externen Sicherheitsmanager empfängt.

Falls RACF als externer Sicherheitsmanager verwendet wird, enthält dieses Feld den RACF-Rückgabecode.

Informationen zu den Werten für ESMRESP finden Sie, wenn Sie den zugehörigen Dokumentationslinks unter ESMREASON folgen.

ISUSERID(datenbereich)

Gibt eine 8 Byte lange Benutzer-ID zurück, die dem Token zugeordnet ist.

OUTTOKEN(zeigerverweis)

Gibt für das Ausgabetoken, das von einem externen Sicherheitsmanager abgerufen wurde, die Adresse im 31-Bit-Taskpeicher zurück.

OUTTOKENLEN(datenbereich)

Gibt die Länge eines Ausgabetokens als Vollwort-Binärfeld zurück.

TOKEN(datenbereich)

Ein Sicherheitstoken mit dem unter TOKENTYPE beschriebenen Typ.

TOKENLEN(datenwert)

Gibt die Länge des Tokens in Form eines Vollwort-Binärwertes an.

TOKENTYPE(cvda)

Gibt den Typ des Tokens an. Die CVDA-Werte lauten wie folgt:

BASICAUTH

Ein Basisauthentifizierungstoken im Format *benutzer-id:kennwort*.

Führende und abschließende Leerzeichen werden für *benutzer-id* und *kennwort* nicht unterstützt. Der Doppelpunkt ist ein erforderliches Trennzeichen. Das Kennwort kann ein Kennwort, eine Kennphrase, ein MFA-Token oder ein PassTicket sein.

JWT

Ein JSON-Web-Token.

KERBEROS

Ein von einem Secure Token Server (STS) abgerufenes Kerberos-Token.

Bedingungen

16 INVREQ

Die RESP2-Werte lauten wie folgt:

13

Der vom externen Sicherheitsmanager zurückgegebene Wert in der Option ESMRESP ist nicht von CICS klassifiziert. Eine Erläuterung der Optionen ESMRESP und ESMREASON finden Sie in der ESM-Dokumentation.

18

Die CICS-Schnittstelle für den externen Sicherheitsmanager ist nicht initialisiert.

29

Der externe Sicherheitsmanager antwortet nicht.

31

Für **TOKENTYPE** wurde ein ungültiger CVDA-Wert angegeben.

32

Für **DATATYPE** wurde ein anderer CVDA-Wert als BASE64 oder BIT angegeben.

36

Der Datentyp BASE64 war angegeben, aber der Wert für **TOKEN** enthält keine Daten in Base64-Codierung.

37

BASE64 ist für diesen Wert für DATATYPE nicht gültig.

40

Das Key-Distribution-Center wurde nicht gestartet oder wird gerade beendet.

41

Das Key-Distribution-Center antwortet nicht.

47

Beim externen Sicherheitsmanager ist keine Benutzer-ID für den Kerberos-Prinzipal definiert, der dem Token zugeordnet ist.

50

Bei den in **TOKEN** angegebenen Daten handelt es sich nicht um ein Kerberos-Token.

51

Das Kerberos-Token unterstützt den Nachrichtenschutz nicht.

52

Das Kerberos-Token gibt an, dass die gegenseitige Authentifizierung genutzt wird, aber die Optionen **OUTTOKEN** und **OUTTOKENLEN** wurden nicht im Befehl angegeben.

53

Diese CICS-Region ist nicht für die Unterstützung von Kerberos konfiguriert. Geben Sie zum Aktivieren der Kerberos-Unterstützung den SIT-Parameter KERBEROSUSER mit der Benutzer-ID an, die dem Serviceprinzipal zugeordnet ist.

60

Das in TOKEN angegebene Token hat ein falsches Format.

62

In ICSF ist ein Fehler aufgetreten.

63

Ein Token wurde aufgrund eines Konfigurationsfehler nicht erstellt.

64

Das MFA-Token ist unvollständig. Es werden nur einzelne Token unterstützt.

65

Der externe Sicherheitsmanager (ESM) unterstützt keine JWTs.

100

Die Klasse IDTDATA ist nicht aktiv.

103

Das JWT ist nicht signiert.

104

ICSF ist nicht verfügbar.

105

Ein ICSF-Fehler ist aufgetreten.

22 LENGERR

Die RESP2-Werte lauten wie folgt:

45

Die Länge des Kerberos-Tokens überschreitet den Maximalwert von 65535.

Standardaktion: Die Task wird abnormal beendet.

70 NOTAUTH

Die RESP2-Werte lauten wie folgt:

20

Der externe Sicherheitsmanager hat die Anforderung zur Verifizierung des Tokens nicht autorisiert. Weitere Informationen enthalten die Fehlnachrichten, die beim CSCS-Ziel geschrieben werden.

42

Eine Kerberos-Anforderung kann nicht ausgeführt werden, weil das zugehörige Ticket abgelaufen ist.

43

Der Authentifikator ist abgelaufen.

61

Das Token wurde nicht akzeptiert.

70

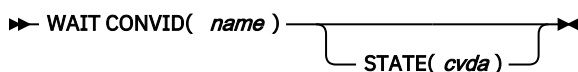
Die Benutzer-ID wurde entzogen.

Standardaktion: Die Task wird abnormal beendet.

WAIT CONVID (APPC)

Stellt sicher, dass kumulierte Daten in einem formatfreien APPC-Datenaustausch übertragen werden.

WAIT CONVID (APPC)



Bedingungen: INVREQ, NOTALLOC

Beschreibung

Mit dem Befehl WAIT CONVID kann ein Anwendungsprogramm sicherstellen, dass alle kumulierten Anwendungsdaten und Steueranzeiger aus einem Befehl SEND oder die Ergebnisse eines Befehls CONNECT PROCESS an die Partnertransaktion übertragen werden.

Optionen

CONVID(name)

Gibt den Dialog an, auf den sich der Befehl bezieht. Der vier Zeichen lange Name gibt entweder das durch einen zuvor ausgeführten Befehl ALLOCATE im Feld EIBRSRCE des EIB zurückgegebene Token

oder aber das Token an, das die Hauptsitzung darstellt (und durch einen zuvor ausgeführten Befehl ASSIGN zurückgegeben wurde).

STATE(*cvda*)

Ruft den Status des aktuellen Datenaustausches ab. Die folgenden CVDA-Werte können von CICS zurückgegeben werden:

- ALLOCATED
- CONFFREE
- CONFRECEIVE
- CONFSEND
- FREE
- PENDFREE
- PENDRECEIVE
- RECEIVE
- ROLLBACK
- SEND
- SYNCFREE
- SYNCRECEIVE
- SYNCSEND

Bedingungen

16 INVREQ

RESP2-Werte:

200

Der Befehl wird für Distributed Program Link nicht unterstützt, wenn er sich auf die Hauptfunktion bezieht.

Diese Bedingung (RESP2 nicht festgelegt) tritt außerdem auf, wenn der Befehl für einen Datenaustausch verwendet wird, der nicht die EXEC CICS-Schnittstelle verwendet oder bei dem es sich nicht um einen formatfreien Datenaustausch handelt.

Standardaktion: Die Task wird abnormal beendet.

61 NOTALLOC

Tritt auf, falls sich der Wert von CONVID nicht auf einen Datenaustausch bezieht, dessen Eigner die Anwendung ist.

Standardaktion: Die Task wird abnormal beendet.

WAIT EVENT

Wartet auf den Eintritt eines Ereignisses.

WAIT EVENT

➤ WAIT EVENT — ECADDR(*zeigerwert*) — NAME(*name*)

Bedingung: INVREQ

Hinweis für dynamisches Transaktionsrouting: Die Verwendung dieses Befehls kann transaktionsübergreifende Affinitäten erzeugen, die die Nutzung des dynamischen Transaktionsroutings beeinträchtigen. Weitere Informationen zu Transaktionsaffinitäten finden Sie unter [Affinität](#).

Beschreibung

Der Befehl WAIT EVENT synchronisiert eine Task mit dem Abschluss eines Ereignisses, das durch dieselbe Task oder durch eine andere Task ausgelöst wurde. Das Ereignis ist normalerweise die Übergabe eines Steuerbereichs für Zeitgeberereignisse, das (zur Ablaufzeit) als Reaktion an einen Befehl POST bereitgestellt wird (eine Beschreibung finden Sie im Abschnitt „POST“ auf Seite 348). Mit dem Befehl WAIT EVENT kann die Steuerung direkt an eine andere Task abgetreten werden, bis das Ereignis, auf das gewartet wird, abgeschlossen ist.

CICS nimmt die Adressen aller Ereignissteuerblöcke, die von Befehlen WAIT EVENT aktueller Tasks übergeben wurden, in das Feld ECBLIST auf, das durch CICS an die Funktion WAIT übergeben wird, wenn keine Arbeit verfügbar ist.

Es sollten gleichzeitig nicht mehrere Tasks auf einen bestimmten Ereignissteuerblock warten. Falls diese Regel nicht befolgt und das durch CICS an die MVS-Funktion WAIT übergebene Feld doppelte Adressen von Ereignissteuerblöcken enthält, bricht MVS CICS ab.

Stellen Sie sicher, dass die asynchrone speicherübergreifende Übergabe (also die Übergabe des Abschlusses für ein Ereignis in einem Adressraum, dessen Eigner nicht der Benutzer ist) nicht häufiger als nötig verwendet wird. Viele speicherübergreifende Übergaben können ein Übermaß an Systemressourcen belegen.

Optionen

ECADDR(zeigerwert)

Gibt die Adresse des Steuerbereichs für Zeitgeberereignisse an, die übergeben werden muss, bevor die Taskaktivität wiederaufgenommen werden kann.

NAME(name)

Gibt den 1 bis 8 alphanumerische Zeichen langen symbolischen Namen an, der in SUSPENDVALUE oder HVALUE zurückgegeben wird, wenn eine Task den Befehl WAIT EVENT ausgibt und Gegenstand eines Befehls INQUIRE TASK oder einer CEMT-Operation INQ TASK ist.

Bedingungen

16 INVREQ

RESP2-Werte:

2

Die Ereignissteuerblockadresse ist ein Nullzeiger (X'00000000' oder X'FF000000').

3

Die angegebene Adresse im Ereignissteuerbereich liegt für Programme, die im 24-Bit-Modus ausgeführt werden, oberhalb der 16-MB-Grenze.

4

Die Adresse im Ereignissteuerbereich ist nicht an einer Vollwortgrenze ausgerichtet.

6

Der im Befehl WAIT EVENT angegebene Steuerbereich für Zeitgeberereignisse befindet sich im Benutzerschlüsselspeicher mit Tasklebensdauer und ist für eine andere Transaktion nicht zugänglich. Diese Bedingung kann nur dann auftreten, wenn der Speicher für den Steuerbereich für Zeitgeberereignisse auf andere Weise als durch einen Befehl POST angefordert wird und zur Übergabe als Ereignissteuerblock durch eine andere Task beim Abschluss eines Ereignisses dient.

Anmerkung: CICS fordert Speicher für einen Steuerbereich für Zeitgeberereignisse als Reaktion auf einen Befehl POST (der zusammen mit dem Befehl WAIT EVENT verwendet werden kann) aus einem gemeinsam genutzten Speicherteilbereich im Benutzerschlüsselspeicher an. Dies stellt sicher, dass sich Steuerbereiche für Zeitgeberereignisse im gemeinsam genutzten Speicher befinden und nicht mit einer Bedingung INVREQ fehlschlagen, wenn in einem nachfolgenden Befehl WAIT EVENT auf sie verwiesen wird.

Standardaktion: Die Task wird abnormal beendet.

Beispiele

Das folgende Beispiel zeigt, wie Sie die Verarbeitung einer Task aussetzen, bis der angegebene Ereignissteuerbereich übergeben wurde:

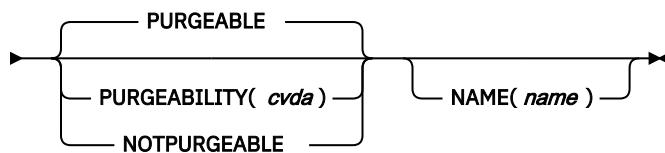
```
EXEC CICS WAIT EVENT ECADDR(PVALUE)
```

WAIT EXTERNAL

Synchronisiert Ereignisse.

WAIT EXTERNAL

►► WAIT EXTERNAL — ECBLIST(*zeigerwert*) — NUMEVENTS(*datenwert*) →



Bedingung: INVREQ

Dieser Befehl ist threadsicher.

Hinweis für dynamisches Transaktionsrouting: Die Verwendung dieses Befehls kann transaktionsübergreifende Affinitäten erzeugen, die die Nutzung des dynamischen Transaktionsroutings beeinträchtigen. Weitere Informationen zu Transaktionsaffinitäten finden Sie unter [Affinität](#).

Beschreibung

Der Befehl WAIT EXTERNAL wartet auf Ereignisse, die Ereignissteuerblöcke im MVS-Format übergeben. Der Befehl bewirkt das Aussetzen der ausgehenden Task, bis einer der Ereignissteuerblöcke übergeben wurde, also bis eines der Ereignisse aufgetreten ist. Die Task kann auf einen oder mehrere Ereignissteuerblöcke warten. Falls sie auf mehrere Ereignissteuerblöcke wartet, ist sie zuteilbar, sobald einer der Ereignissteuerblöcke übergeben wird. Sie müssen sicherstellen, dass jeder Ereignissteuerblock spätestens zum frühestmöglichen Zeitpunkt seiner Übergabe gelöscht (= auf binäre Nullen gesetzt) wird. Diese Aufgabe kann nicht von CICS für Sie übernommen werden. Falls Sie auf einen Ereignissteuerblock warten, der zuvor übergeben und anschließend nicht gelöscht wurde, wird Ihre Task nicht ausgesetzt und weiter so ausgeführt, als ob WAIT EXTERNAL nicht ausgegeben wurde.

CICS verwendet erweiterte Ereignissteuerblöcke und MVS-Exitmechanismen POST für Ereignissteuerblöcke, die durch einen Befehl WAIT EXTERNAL übergeben werden. Verwenden Sie daher den Befehl WAIT EXTERNAL nur dann, wenn Sie sicher sind, dass die Ereignissteuerblöcke nicht durch eine andere Methode als den MVS-Service POST oder die Standardlogik für die 'optimierte Übergabe' unter Verwendung einer Anweisung zum Vergleichen und Auslagern (Compare-and-swap, CS) übergeben werden. Die Standardlogik für die 'optimierte Übergabe' ist übrigens nur dann anwendbar, wenn sich der Ereignissteuerblock nicht im Wartestatus befindet, das Wartebit X'80' also nicht aktiviert ist.

Falls ein Ereignissteuerblock für einen Befehl WAIT EXTERNAL manuell übergeben wird, weil beispielsweise eine andere Task einen Wert in den Ereignissteuerblock versetzt, können unvorhersehbare Fehler auftreten. Verwenden Sie den Befehl WAITCICS, falls die Möglichkeit besteht, dass manuelle Übergaben stattfinden. Verwenden Sie den Befehl WAIT EXTERNAL, wann immer dies möglich ist, da er normalerweise mit geringerem Systemaufwand verbunden ist.

Es dürfen gleichzeitig nicht mehrere Tasks auf einen bestimmten Ereignissteuerblock warten. Falls diese Regel nicht befolgt wird, empfängt die zweite auf den Ereignissteuerblock wartende Task eine Bedingung INVREQ.

Stellen Sie sicher, dass die asynchrone speicherübergreifende Übergabe (also die Übergabe des Abschlusses für ein Ereignis in einem Adressraum, dessen Eigner nicht der Benutzer ist) nicht häufiger als nötig verwendet wird. Viele speicherübergreifende Übergaben können ein Übermaß an Systemressourcen belegen.

Optionen

ECBLIST(*zeigerwert*)

Ist ein Zeiger auf eine Liste der Adressen von Ereignissteuerblöcken im MVS-Format, die Ereignisse darstellen. Sowohl das Feld ECBLIST als auch die Ereignissteuerblöcke können oberhalb der 16-MB-Grenze liegen, also 31-Bit-Adressen sein. Jeder Ereignissteuerblock muss am Vollwort ausgerichtet sein. Adressen von Ereignissteuerblöcken, bei denen es sich um Nullwerte handelt (X'00000000' und X'FF000000') werden ignoriert.

NAME(*name*)

Gibt den 1 bis 8 alphanumerische Zeichen langen symbolischen Namen an, der in SUSPENDVALUE oder HVALUE zurückgegeben wird, wenn eine Task den Befehl WAIT EXTERNAL ausgibt und Gegenstand eines Befehls INQ TASK oder einer CEMT-Operation INQ TASK ist.

NUMEVENTS(*datenwert*)

Gibt die Anzahl solcher Ereignisse an und entspricht der Anzahl von Adressen im Feld ECBLIST. Dieses Feld enthält eine Vollwort-Binärzahl. Wenn für NUMEVENTS der Wert 1 angegeben ist, muss das Feld ECBLIST trotzdem eine Adresse enthalten, die auf eine Liste verweist, in der lediglich eine einzige Ereignissteuerblockadresse angegeben ist.

PURGEABILITY(*cvda*)

Bestimmt das Ergebnis

- bei der versuchten Ausführung eines Deadlock-Zeitlimits
- bei einem Befehl SET TASK PURGE|FORCEPURGE
- bei einer CEMT-Operation SET TASK PURGE|FORCEPURGE

für die ausgebende Task, während sich diese im Wartestatus befindet. Die Werte werden an CICS im Feld PURGEABLE (Standardwert) oder im Feld NOTPURGEABLE übergeben. Das Ergebnis ist Folgendes:

Funktion	PURGEABLE	NOTPURGEABLE
DTIMOUT abgelaufen	Abbruch von AEXY	Keine Auswirkung
CEMT SET TASK PURGE EXEC CICS SET TASK PURGE	Abbruch von AEXY	Keine Auswirkung
CEMT SET TASK FORCEPURGE EXEC CICS SET TASK FORCEPURGE	Abbruch von AEXY	Abbruch von AEXY

Im Abschnitt [Fehlerbehebung bei Transaktionen](#) finden Sie weitere Informationen zu DTIMOUT und SET TASK PURGE|FORCEPURGE.

Bedingungen

16 INVREQ

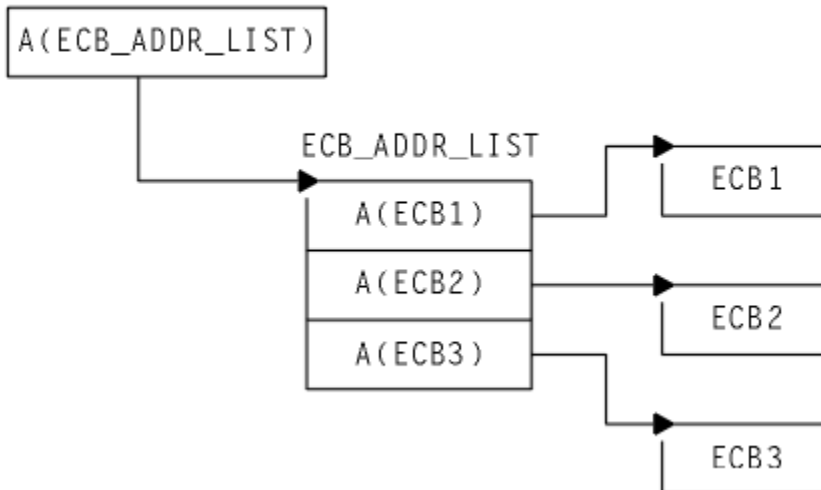
Die RESP2-Werte lauten wie folgt:

- 1** Ein Ereignissteuerblock ist nicht gültig, beispielsweise weil er nicht an einem Vollwort ausgerichtet ist.
 - 2** Die Ereignissteuerblockadresse ist ein Nullzeiger (X'00000000' oder X'FF000000').
 - 3** Der Wert für NUMEVENTS ist keine positive Zahl.
 - 4** Die Option PURGEABILITY wurde mit einem falschen CVDA-Wert angegeben.
 - 5** In der Liste wurden keine gültigen Ereignissteuerblöcke gefunden, weil entweder die in ECBLIST angegebene Adresse nicht gültig ist oder weil alle Ereignissteuerblockadressen nicht gültig sind.
Die angegebenen Ereignissteuerblöcke befinden sich im Nur-Lese-Speicher.
- Standardaktion: Die Task wird abnormal beendet.

Beispiele

Die folgende Abbildung zeigt, wie Sie mit dem Parameter ECBLIST auf eine Liste von Ereignissteuerblockadressen verweisen, die wiederum auf einzelne Ereignissteuerblöcke verweisen. Bitte beachten Sie, dass die Variable ECBLIST ein Zeiger ist, der auf die erste Adresse in der Liste verweist.

PTR_ECB_ADDR_LIST



```

DCL
  ECB1      FIXED BIN(31),      /* actual ecb */
  ECB2      FIXED BIN(31),      /* actual ecb */
  ECB3      FIXED BIN(31);      /* actual ecb */
DCL          /* list of ecb addresses */
  1  ECB_ADDR_LIST,
  2  ECB_ADDR(3)  PTR;
DCL          /* ptr to each addr list */
  PTR_ECB_ADDR_LIST  PTR;
ECB_ADDR(1) = ADDR(ECB1);
ECB_ADDR(2) = ADDR(ECB2);
ECB_ADDR(3) = ADDR(ECB3);
/* set up pointer */
PTR_ECB_ADDR_LIST = ADDR(ECB_ADDR_LIST);
/* PTR_ECB_ADDR_LIST = ADDR(ECB_ADDR(1));
   (alternative) */
EXEC CICS WAIT  EXTERNAL
               ECBLIST(PTR_ECB_ADDR_LIST)
               NUMEVENTS(3)
               PURGEABLE
  
```

Abbildung 3. Option ECBLIST beim EXEC CICS-Befehl WAIT EXTERNAL

WAIT JOURNALNAME

Synchronisiert die Task mit der Journalausgabe, die in den Protokolldatenstrom einer Systemprotokollfunktion geschrieben wird.

WAIT JOURNALNAME

➡ WAIT JOURNALNAME(*datenwert*) ———— REQID(*datenwert*) ————

Bedingungen: IOERR, JIDERR, NOTOPEN

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl WAIT JOURNALNAME synchronisiert die Task mit der Ausgabe von einem oder mehreren Journalsätzen, die zwar erstellt wurden, aber deren Ausgabe verzögert wurde; die Synchronisierung erfolgt also mit asynchronen Journalausgabeanforderungen.

Die Journalsätze können bereits aus dem Journalpufferbereich in den Protokolldatenstrom der entsprechenden Systemprotokollfunktion geschrieben worden sein oder die Ausgabeoperation der Systemprotokollfunktion kann gegenwärtig noch verarbeitet werden. Falls die Ausgabeoperation für den Protokolldatenstrom bereits abgeschlossen wurde, wird die Steuerung sofort an die anfordernde Task zurückgegeben. Andernfalls wartet die anfordernde Task, bis die Operation abgeschlossen worden ist.

Falls das aufrufende Programm eine Folge von erfolgreichen Anforderungen für die asynchrone Ausgabe an dasselbe Journal ausgegeben hat, muss lediglich die letzte dieser Anforderungen synchronisiert werden, um sicherzustellen, dass alle Journalsätze an den Protokolldatenstrom der Systemprotokollfunktion ausgegeben wurden. Hierzu kann entweder ein eigenständiger Befehl WAIT JOURNALNAME ausgegeben oder der letzte Ausgabebefehl selbst als synchron definiert werden (durch Angabe der Option WAIT im Befehl WRITE JOURNALNAME).

Optionen

JOURNALNAME(datenwert)

Gibt einen 1 bis 8 Zeichen langen Journalnamen an, um das Journal zu kennzeichnen, auf dessen Synchronisierung die Task wartet. Der Name muss ein CICS bekannter Journalname sein.

Um den Wartestatus für das CICS-Systemprotokoll auszugeben, geben Sie DFHLOG als Journalnamen an.

Um den Wartestatus für Journale auszugeben, die mit der Konvention für die Journalnummerierung definiert sind (z. B. in FILE-Ressourcendefinitionen), geben Sie den Namen als DFHJnn an, wobei 'nn' für die Journalnummer im Bereich von 1 bis 99 steht.

Anmerkung: Die Angabe von DFHJ01 bei diesem Befehl bezieht sich auf ein Benutzerjournal und *nicht* auf das Systemprotokoll.

REQID(datenwert)

Gibt als Vollwort-Binärzahlvariable das Token an, das sich auf einen Journalsatz bezieht, der erstellt, aber möglicherweise noch nicht ausgegeben wurde. Das Token wurde von CICS für einen vorherigen Befehl WRITE JOURNALNAME zurückgegeben, der durch diese Task ausgegeben wurde.

Falls Sie die Option REQID nicht angeben, wird die Task mit der Ausgabe des aktuellen Puffers für das Journal synchronisiert, das durch die Option JOURNALNAME angegeben wird.

Bedingungen

17 IOERR

Ein Journalsatz wurde nicht ausgegeben, weil die Systemprotokollfunktion oder SMF eine Bedingung für einen nicht behebbaren Fehler zurückgegeben hat.

Standardaktion: Falls es sich bei dem Protokoll um das Systemprotokoll handelt, wird CICS entweder in den Quiesce-Status versetzt oder abgebrochen. Handelt es sich um ein allgemeines Protokoll, wird die Task abnormal beendet.

43 JIDERR

Tritt in einer der folgenden Situationen auf:

- Der angegebene Journalname ist in der CICS-Region nicht bekannt.
- Der angegebene Journalname bezieht sich auf einen reinen DASD-Protokolldatenstrom, mit dem gegenwärtig eine CICS-Region in einem anderen MVS-Image verbunden ist.

Standardaktion: Die Task wird abnormal beendet.

19 NOTOPEN

Tritt in einer der folgenden Situationen auf:

- Der Befehl kann nicht ausgeführt werden, weil der Benutzer das angegebene Journal explizit inaktiviert hat.
- Es wurde eine Warteanforderung für ein Journal ausgegeben, in das zuvor keine Daten geschrieben wurden.
- Dieses Journal wurde unter Verwendung eines Modells definiert, das es in dem Protokolldatenstrom zuordnet, der durch das Systemprotokoll für dieses System verwendet wird. Dieser Fehler wird bei dem Versuch, eine Verbindung zum Protokolldatenstrom herzustellen, nicht erkannt. Für das Journal wird eine Definition installiert und auf einen fehlgeschlagenen Status gesetzt.

Standardaktion: Die Task wird abnormal beendet.

70 NOTAUTH

Eine Ressourcensicherheitsprüfung für den Wert von JOURNALNAME(datenwert) ist fehlgeschlagen.

Standardaktion: Die Task wird abnormal beendet.

Beispiele

Das folgende Beispiel zeigt, wie Sie die Synchronisierung mit der Ausgabe der Journalsätze anfordern, die in ein Benutzerjournal namens 'ACCOUNTS' geschrieben wurden:

```
EXEC CICS WAIT JOURNALNAME('ACCOUNTS')
      REQID(RECTOKEN)
```

WAIT JOURNALNUM

Führt eine Synchronisierung mit der Journalausgabe durch.

Dieser Befehl wird aus Gründen der Kompatibilität mit früheren Releases von CICS unterstützt. Er wurde durch den Befehl WAIT JOURNALNAME ersetzt, der stattdessen verwendet werden sollte.

Die Syntax ist dieselbe wie beim Befehl WAIT JOURNALNAME; eine Ausnahme besteht lediglich darin, dass JOURNALNUM anstelle eines Zeichenwertes einen numerischen Wert angibt. Der numerische Wert 'nn' liegt im Bereich zwischen 01 und 99 und entspricht dem Journalnamen DFHJnn.

Dieser Befehl ist threadsicher.

WAIT SIGNAL

Setzt eine Task für eine logische Einheit aus.

WAIT SIGNAL

➡ WAIT SIGNAL ➡

Bedingungen: NOTALLOC, SIGNAL, TERMERR

Beschreibung

Der Befehl WAIT SIGNAL setzt - nur für eine Hauptfunktion - eine Task aus, bis eine Bedingung SIGNAL auftritt. Einige logische Einheiten können den normalen Datenfluss an das Anwendungsprogramm durch einen Datenflusssteuerbefehl SIGNAL an CICS unterbrechen und einen Abruf signalisieren, der wiederum das Auftreten der Bedingung SIGNAL zur Folge hat.

Der Befehl HANDLE CONDITION SIGNAL bewirkt eine Verzweigung zu einer Benutzerroutine, wenn ein Abruf empfangen wird.

Ein Befehl WAIT SIGNAL kann für die folgenden logischen Einheiten codiert werden:

- LUTYPE4
- LU-Typ 6.1
- 3600 (3601)
- Interaktive logische 3767-Einheit
- 3770, Stapelverarbeitung
- Logische 3790-Einheit mit vollem Funktionsumfang

Bedingungen

61 NOTALLOC

Tritt auf, falls es sich bei der Hauptfunktion der Task nicht um ein Terminal handelt.

Standardaktion: Die Task wird abnormal beendet.

24 SIGNAL

Tritt auf, wenn der Datenflusssteuerbefehl von der Hauptfunktion empfangen wurde.

Beim Empfang eines eingehenden Signals wird das Feld EIBSIG immer festgelegt.

Standardaktion: Die Bedingung wird ignoriert.

81 TERMERR

Tritt für einen terminalbezogenen Fehler auf.

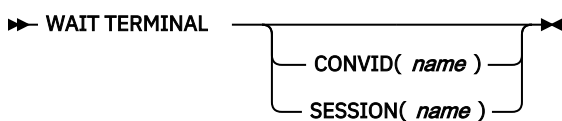
Diese Bedingung kann durch eine Anforderung CANCEL TASK durch ein Knotenfehlerprogramm des Benutzers verursacht werden, falls für die Task eine Terminalsteuerungsanforderung aussteht, wenn der Sitzungsfehler durch das Programm des Knotens für abnormale Bedingungen (CSNE) abgewickelt wird.

Standardaktion: Die Task wird abnormal mit dem Abbruchcode ATNI beendet.

WAIT TERMINAL

Stellt sicher, dass die Terminaloperation abgeschlossen wurde.

WAIT TERMINAL



Bedingungen: EOC, INVREQ, NOTALLOC, SIGNAL, TERMERR

Beschreibung

Der Befehl WAIT TERMINAL stellt sich, dass eine Terminaloperation abgeschlossen wurde.

Optionen

CONVID(name)

Gibt den Dialog an, auf den sich der Befehl bezieht. Der vier Zeichen lange Name bezeichnet entweder das Token, das von einem zuvor ausgeführten Befehl ALLOCATE in EIBRSRCE im EIB zurückgegeben wurde, oder das Token, das die Hauptsitzung (die durch einen zuvor ausgeführten Befehl ASSIGN zurückgegeben wurde) darstellt.

SESSION(name)

Gibt die symbolische ID (1-4 Zeichen) für einen Terminaleintrag in der Terminalsteuertabelle (TCTTE) für die Sitzung an. Diese Option gibt die zu verwendende alternative Funktion an. Falls sowohl diese Option als auch die Option CONVID nicht angegeben ist, wird die Hauptfunktion für die Task verwendet.

Bedingungen

06 EOC

Tritt auf, wenn die Anforderung/Antwort-Einheit mit festgelegtem Anzeiger für das Kettenende empfangen wird. Das Feld EIBEOC gibt diese Bedingung ebenfalls an.

Standardaktion: Die Bedingung wird ignoriert.

16 INVREQ

RESP2-Werte:

200

Eine Serveranwendung für Distributed Program Link hat in der Option CONVID die Sitzung für die Funktionsverlagerung (ihre Hauptfunktion) angegeben.

Standardaktion: Die Task wird abnormal beendet.

61 NOTALLOC

Tritt auf, falls die Anwendung nicht Eigner der im Befehl angegebenen Funktion ist.

Standardaktion: Die Task wird abnormal beendet.

24 SIGNAL

Tritt auf, wenn ein eingehender Befehl SIGNAL für die Datenflusssteuerung von einer logischen Einheit oder Sitzung empfangen wird. Beim Empfang eines eingehenden Signals wird das Feld EIBSIG immer festgelegt.

Standardaktion: Die Bedingung wird ignoriert.

81 TERMERR

Tritt für einen terminalbezogenen Fehler auf.

Diese Bedingung kann durch eine Anforderung CANCEL TASK durch ein Knotenfehlerprogramm des Benutzers verursacht werden, falls für die Task eine Terminalsteuerungsanforderung aussteht, wenn der Sitzungsfehler durch das Programm des Knotens für abnormale Bedingungen abgewickelt wird.

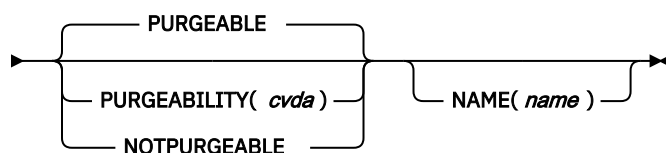
Standardaktion: Die Task wird abnormal mit dem Abbruchcode ATNI beendet.

WAITCICS

Synchronisiert Ereignisse.

WAITCICS

➡ WAITCICS — ECBLIST(*zeigerwert*) — NUMEVENTS(*datenwert*) —➡



Bedingung: INVREQ

Hinweis für dynamisches Transaktionsrouting: Die Verwendung dieses Befehls kann transaktionsübergreifende Affinitäten erzeugen, die die Nutzung des dynamischen Transaktionsroutings beeinträchtigen. Weitere Informationen zu Transaktionsaffinitäten finden Sie unter [Affinität](#).

Beschreibung

Der Befehl WAITCICS wartet auf Ereignisse, die Ereignissteuerblöcke im MVS-Format übergeben. Der Befehl bewirkt das Aussetzen der ausgebenden Task, bis einer der Ereignissteuerblöcke übergeben wurde, also bis eines der Ereignisse aufgetreten ist. Die Task kann auf einen oder mehrere Ereignissteuerblöcke warten. Falls sie auf mehrere Ereignissteuerblöcke wartet, ist sie zuteilbar, sobald einer der Ereignissteuerblöcke übergeben wird. Sie müssen sicherstellen, dass jeder Ereignissteuerblock spätestens zum frühestmöglichen Zeitpunkt seiner Übergabe gelöscht (= auf binäre Nullen gesetzt) wird. Diese Aufgabe kann nicht von CICS für Sie übernommen werden. Falls Sie auf einen Ereignissteuerblock warten, der zuvor übergeben und anschließend nicht gelöscht wurde, wird Ihre Task nicht ausgesetzt und weiter so ausgeführt, als ob WAITCICS nicht ausgegeben wurde.

CICS nimmt die Adressen aller Ereignissteuerblöcke, die von Befehlen WAITCICS aktueller Tasks übergeben wurden, in das Feld ECBLIST auf, das durch CICS an die MVS-Funktion WAIT übergeben wird, wenn keine Arbeit verfügbar ist. Solche Ereignissteuerblöcke können mit der MVS-Funktion POST oder durch eine manuelle Übergabe übergeben werden. Die manuelle Übergabe könnte beispielsweise durch das Versetzen eines entsprechenden Wertes in den Ereignissteuerblock erfolgen. Falls die manuelle Übergabe definitiv nicht verwendet wird, sollte besser WAIT EXTERNAL verwendet werden.

Es sollten gleichzeitig nicht mehrere Tasks auf einen bestimmten Ereignissteuerblock warten. Falls diese Regel nicht befolgt und das durch CICS an die MVS-Funktion WAIT übergebene Feld doppelte Adressen von Ereignissteuerblöcken enthält, bricht MVS CICS ab.

Optionen

ECBLIST(zeigerwert)

Ist ein Zeiger auf eine Liste der Adressen von Ereignissteuerblöcken im MVS-Format, die Ereignisse darstellen. Sowohl das Feld ECBLIST als auch die Ereignissteuerblöcke können oberhalb der 16-MB-Grenze liegen, also 31-Bit-Adressen sein. Jeder Ereignissteuerblock muss am Vollwort ausgerichtet sein. Adressen von Ereignissteuerblöcken, bei denen es sich um Nullwerte handelt (X'00000000' und X'FF000000') werden ignoriert.

NAME(name)

Gibt den 1 bis 8 alphanumerische Zeichen langen symbolischen Namen als Ursache für den Wartestatus an. Der von Ihnen angegebene Wert wird in der Option SUSPENDVALUE bzw. HVALUE der EXEC CICS-Befehle INQ TASK oder CEMT INQ TASK zurückgegeben.

NUMEVENTS(datenwert)

Gibt die Anzahl solcher Ereignisse an und entspricht der Anzahl von Adressen im Feld ECBLIST. Dieses Feld enthält eine Vollwort-Binärzahl. Wenn für NUMEVENTS der Wert 1 angegeben ist, muss das Feld ECBLIST trotzdem eine Adresse enthalten, die auf eine Liste verweist, in der lediglich eine einzige Ereignissteuerblockadresse angegeben ist.

PURGEABILITY(cvda)

Bewirkt das Aussetzen der ausgebenden Task, bis einer der Ereignissteuerblöcke übergeben wurde, bis also eines der Ereignisse aufgetreten ist. Die Werte werden an CICS im Feld PURGEABLE (Standardwert) oder im Feld NOTPURGEABLE übergeben. Dieses Feld enthält eine Vollwort-Binärzahl. Falls eine andere Funktion versucht, das Feld zu löschen, während sich diese Task im Wartestatus befindet, hat dies folgendes Ergebnis:

Funktion	PURGEABLE	NOTPURGEABLE
DTIMOUT abgelaufen	Abbruch von AEXY	Keine Auswirkung
CEMT SET TASK PURGE EXEC CICS SET TASK PURGE	Abbruch von AEXY	Keine Auswirkung

Funktion	PURGEABLE	NOTPURGEABLE
CEMT SET TASK FORCEPURGE EXEC CICS SET TASK FORCEPURGE	Abbruch von AEXY	Abbruch von AEXY

Informationen zu DTIMOUT finden Sie unter Fehlerbehebung [bei Transaktionen](#); Angaben über SET TASK PURGE|FORCEPURGE enthält der Abschnitt [SET TASK](#).

Bedingungen

16 INVREQ

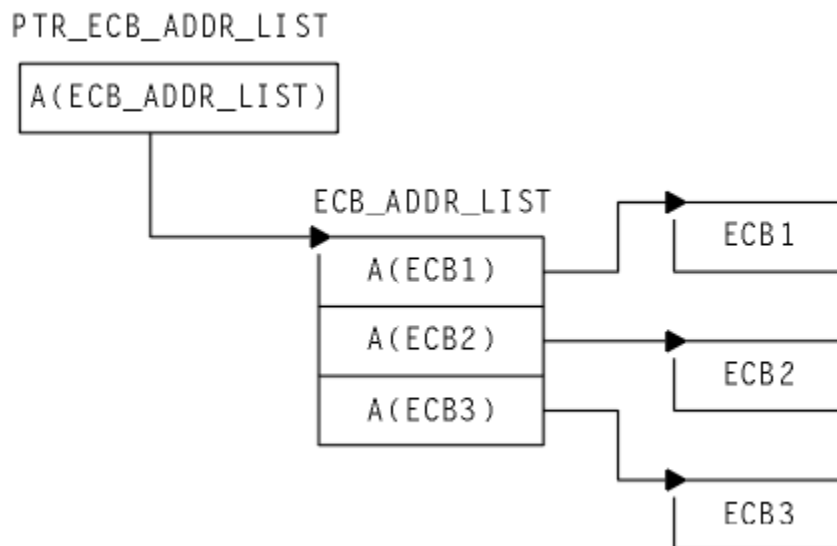
RESP2-Werte:

- 1** Ein Ereignissteuerblock ist nicht gültig, beispielsweise weil er nicht an einem Vollwort ausgerichtet ist.
- 3** Der Wert für NUMEVENTS ist keine positive Zahl.
- 4** Die Option PURGEABILITY wurde mit einem falschen CVDA-Wert angegeben.
- 5** In der Liste wurden keine gültigen Ereignissteuerblöcke gefunden, weil entweder die in ECBLIST angegebene Adresse nicht gültig ist oder weil alle Ereignissteuerblockadressen nicht gültig sind.
Die angegebenen Ereignissteuerblöcke befinden sich im Nur-Lese-Speicher.

Standardaktion: Die Task wird abnormal beendet.

Beispiele

Die folgende Abbildung zeigt, wie Sie mit dem Parameter ECBLIST auf eine Liste von Ereignissteuerblockadressen verweisen, die wiederum auf einzelne Ereignissteuerblöcke verweisen. Bitte beachten Sie, dass die Variable ECBLIST ein Zeiger ist, der auf die erste Adresse in der Liste verweist.



```

DCL
  ECB1      FIXED BIN(31),      /* actual ecb */
  ECB2      FIXED BIN(31),      /* actual ecb */
  ECB3      FIXED BIN(31);      /* actual ecb */
DCL          /* list of ecb addresses */
  1  ECB_ADDR_LIST,
  2  ECB_ADDR(3)  PTR;
DCL          /* ptr to each addr list */
  PTR_ECB_ADDR_LIST  PTR;
ECB_ADDR(1) = ADDR(ECB1);
ECB_ADDR(2) = ADDR(ECB2);
ECB_ADDR(3) = ADDR(ECB3);
/* set up pointer */
PTR_ECB_ADDR_LIST = ADDR(ECB_ADDR_LIST);
/* PTR_ECB_ADDR_LIST = ADDR(ECB_ADDR(1));
   (alternative) */
EXEC CICS WAITCICS
      ECBLIST(PTR_ECB_ADDR_LIST)
      NUMEVENTS(3)
      PURGEABLE
  
```

Abbildung 4. Option ECBLIST beim EXEC CICS-Befehl WAITCICS

WEB CLOSE

Beendet die Verwendung einer Verbindung zu einem Server für CICS als HTTP-Client durch eine Anwendung.

WEB CLOSE

➡ WEB — CLOSE — SESSTOKEN(*datenwert*) ➡

Bedingungen: NOTOPEN

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl WEB CLOSE benachrichtigt CICS darüber, dass das Anwendungsprogramm die Verwendung einer HTTP-Clientverbindung zu einem Server beendet hat. Das Sitzungstoken stellt die Nutzung der Verbindung durch die Anwendung dar. Nachdem die Anwendung den Befehl WEB CLOSE ausgegeben hat, ist die Verwendung des angegebenen Sitzungstokens nicht mehr gültig. Das Sitzungstoken ist erforderlich, damit eine Antwort vom Server empfangen und die HTTP-Header für die Antwort gelesen werden können,

daher sollte der Befehl WEB CLOSE erst nach Abschluss aller Interaktionen mit dem Server und der von ihm gesendeten Antwort ausgegeben werden.

Wenn das Anwendungsprogramm die Verwendung einer HTTP-Clientverbindung beendet, kann CICS die Verbindung schließen oder auch nicht:

- Falls die Verbindung noch geöffnet ist, wenn Sie den Befehl WEB CLOSE absetzen, und sie mit einer URIMAP-Ressource geöffnet wurde, die das Verbindungspooling angibt, schließt CICS die Verbindung nicht. CICS überprüft den Status der Verbindung und stellt sie für eine Wiederverwendung in den Pool.
- Falls die Verbindung bei Ausgabe des Befehls WEB CLOSE nicht für das Verbindungspooling geeignet ist, weil der Server oder Ihr Anwendungsprogramm das Schließen der Verbindung angefordert hat, weil die Verbindung nicht unter Verwendung einer geeigneten URIMAP-Ressource geöffnet wurde oder weil die Verbindung keinen fehlerfreien Zustand aufweist, schließt CICS die Verbindung und stellt sie nicht in einen Pool.
- Falls Sie den Befehl WEB CLOSE nicht ausgeben, schließt CICS die Verbindung am Ende der Task. Die geschlossene Verbindung kann nicht in einen Pool gestellt werden. Um das Verbindungspooling zu aktivieren, muss Ihre Anwendung den Befehl WEB CLOSE ausgeben.

Wenn Sie den Befehl WEB CLOSE ausgeben, fordert CICS nicht den Server zum Schließen der Verbindung auf. Falls Sie kein Verbindungspooling verwenden und CICS die Verbindung nach der Beendigung des Anwendungsprogramms schließt, hat sich das Verfahren bewährt, durch Angabe der Option CLOSESTATUS(CLOSE) im abschließenden Befehl WEB SEND oder WEB CONVERSE das Schließen der Verbindung vom Server anzufordern. Bei Angabe dieser Option schreibt CICS einen Header 'Connection: close' in die Anforderung bzw. lässt bei einem Server mit HTTP-Version 1.0 den Header 'Connection: Keep-Alive' weg. Die Informationen in den Headern bedeuten, dass der Server seine Verbindung unmittelbar nach dem Senden der letzten Antwort schließen kann und nicht auf weitere Anforderungen warten muss, bevor das Zeitlimit abläuft. Geben Sie bei Verwendung des Verbindungspoolings die Option CLOSESTATUS(CLOSE) nicht an, da geschlossene Verbindungen nicht in den Pool gestellt werden können.

Die Verbindung kann auf Anforderung des Servers geschlossen werden, bevor der Befehl WEB CLOSE ausgegeben wird. Falls Sie testen müssen, ob der Server das Schließen der Verbindung angefordert hat, suchen Sie mit dem Befehl WEB READ HTTPHEADER in der letzten Nachricht vom Server nach dem Header 'Connection: close'.

Falls der Server das Schließen der Verbindung anfordert, bleiben die mit dieser Verbindung zusammenhängenden Daten noch in CICS verfügbar, bis der Befehl WEB CLOSE ausgegeben wird. Zu den verfügbaren Daten gehören die letzte vom Server empfangene Nachricht und die Parameter, die zum Öffnen der Verbindung verwendet wurden (z. B. das Schema und der Hostname des Servers). Sobald ein Server die Verbindung geschlossen hat, kann das Anwendungsprogramm die folgenden Tasks nicht mehr ausführen:

- Weitere Anforderungen mit den Befehlen WEB SEND oder WEB CONVERSE über diese Verbindung senden
- HTTP-Header mit dem Befehl WEB WRITE HTTPHEADER schreiben

Das Anwendungsprogramm kann jedoch weiterhin die folgenden Tasks ausführen:

- Antwort mit dem Befehl WEB RECEIVE empfangen
- HTTP-Header mit dem Befehl WEB READ HTTPHEADER und dem Suchbefehl für HTTP-Header prüfen
- Verbindungsinformationen mit dem Befehl WEB EXTRACT extrahieren

Sobald der Befehl WEB CLOSE ausgegeben wird, werden die Daten, die mit der Nutzung der Verbindung durch die Anwendung zusammenhängen, gelöscht.

Optionen

SESTOKEN(datenwert)

Gibt den 8 Byte umfassenden Binärwert für das Sitzungstoken an, der die Nutzung einer HTTP-Clientverbindung zwischen CICS und einem Server durch eine Anwendung eindeutig kennzeichnet. Dieser Wert wird durch einen Befehl WEB OPEN für CICS als HTTP-Client zurückgegeben. Wenn Sie den Befehl WEB CLOSE für die durch das Sitzungstoken gekennzeichnete Sitzung ausgeben, löscht CICS die Daten, die mit der Nutzung der Verbindung durch die Anwendung zusammenhängen, und macht das

Sitzungstoken zur weiteren Verwendung durch das Anwendungsprogramm ungültig. Die Verwendung des Sitzungstokens ist im Abschnitt Sitzungstokens erläutert.

Bedingungen

19 NOTOPEN

Die RESP2-Werte lauten wie folgt:

27

Das Sitzungstoken ist ungültig.

144

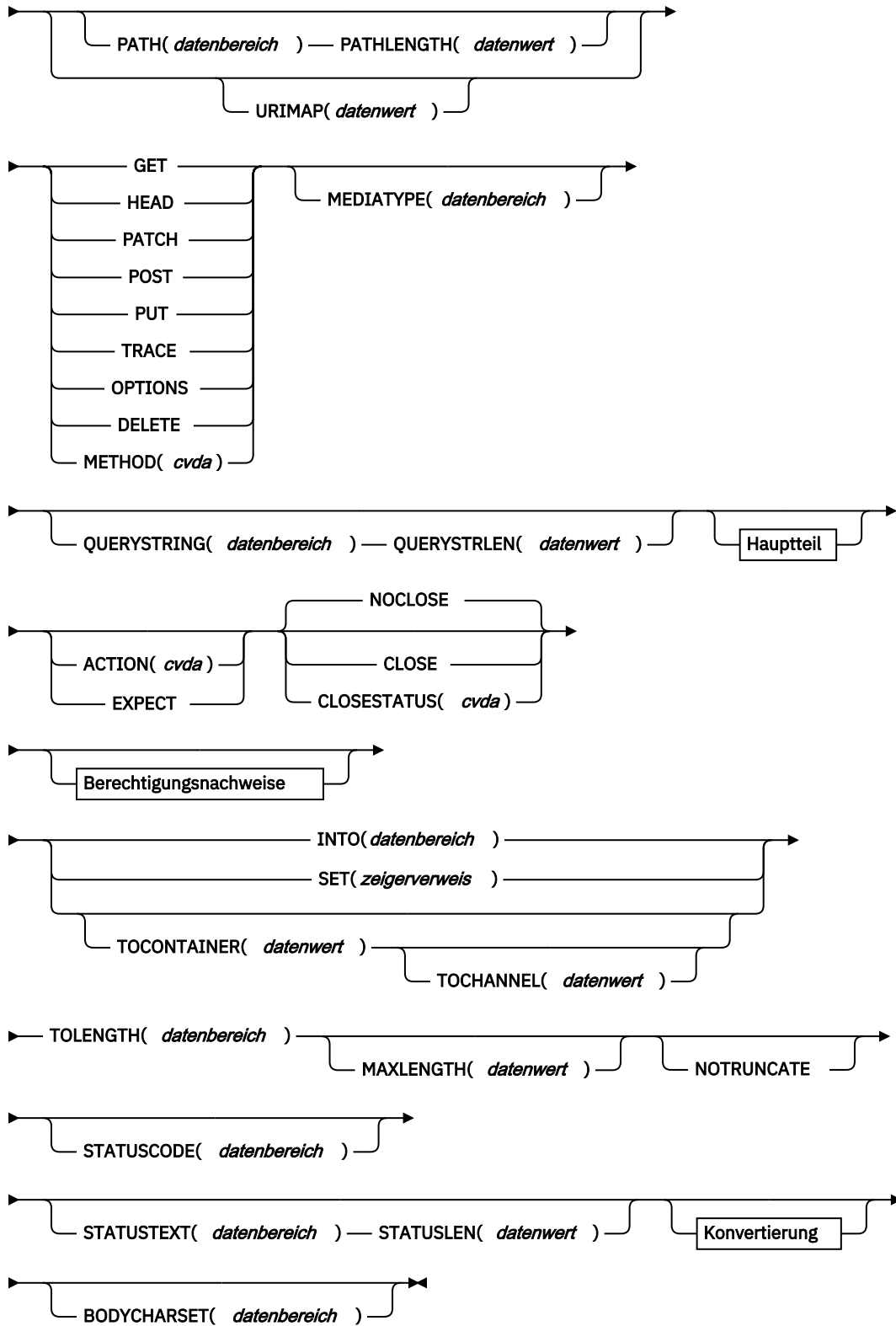
Einer oder mehrere Parameter des Webbefehls sind ungültig.

WEB CONVERSE

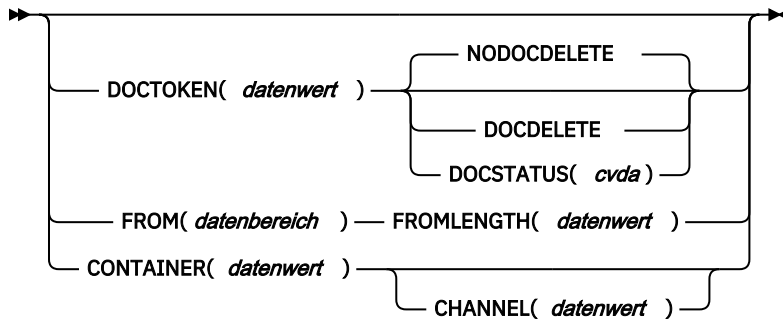
Sendet eine HTTP-Anforderung durch CICS als HTTP-Client und empfängt eine Antwort vom Server. Beide Aktionen finden mithilfe eines einzigen Befehls statt. Der Befehl WEB CONVERSE ist eine Alternative zu den Befehlen WEB SEND und WEB RECEIVE für CICS als HTTP-Client.

WEB CONVERSE

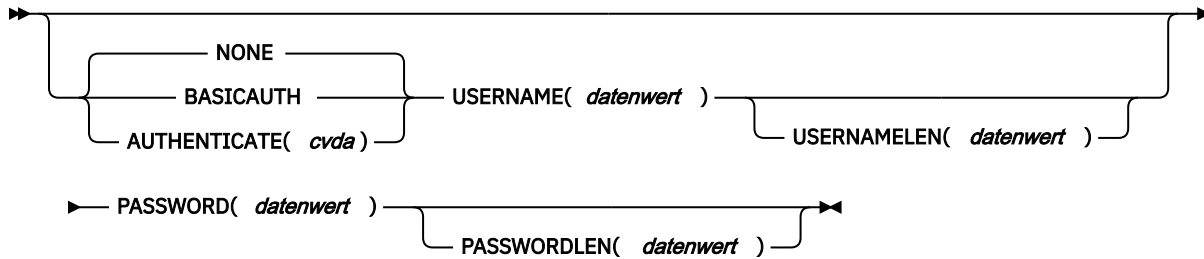
►► WEB — CONVERSE — SESSTOKEN(*datenwert*) →



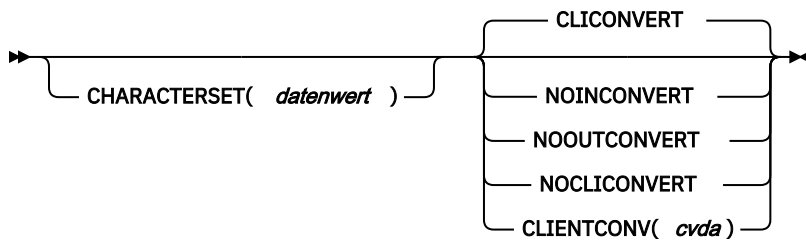
Hauptteil



Berechtigungsnachweise



Konvertierung



Bedingungen: CHANNELERR, CONTAINERERR, IOERR, INVREQ, LENGERR, NOTAUTH, NOTFND, NOT-OPEN, TIMEOUT, TOKENERR

Dieser Befehl ist threadsicher.

Beschreibung

Mit dem Befehl WEB CONVERSE kann ein Anwendungsprogramm eine HTTP-Clientanforderung erstellen und senden sowie eine Antwort vom Server empfangen. In diesen Befehl muss ein Sitzungstoken einbezogen werden. Anleitungen für die korrekte Verwendung des Befehls WEB CONVERSE enthält der Abschnitt [HTTP-Anforderungen über CICS als HTTP-Client absetzen](#).

- Die **HTTP-Clientanforderung** wird mithilfe einer Verbindung ausgegeben, die durch den Befehl WEB OPEN geöffnet wurde. Der Befehl WEB CONVERSE kann anstelle des Befehls WEB SEND verwendet werden, um die Anforderung zu erstellen und zu senden.
- Die **Antwort vom Server** wird von der CICS-Webunterstützung empfangen und an die Anwendung übergeben. Der Befehl WEB CONVERSE kann anstelle des Befehls WEB RECEIVE verwendet werden, damit das Anwendungsprogramm auf die HTTP-Antwort wartet und sie empfängt. Die Header für die HTTP-Antwort können mit dem Befehl WEB READ HTTPHEADER oder den Suchbefehlen für HTTP-Header separat geprüft werden.

Anmerkung: Der Wert von RTIMOUT, der für die Transaktion angegeben ist, von der die Benutzeranwendung gestartet wird, gibt die Zeit an, die die Anwendung für das Warten auf den Empfang der eingehenden Nachricht vorbereitet ist. (Der Wert von RTIMOUT ist in der Definition des Transaktionsprofils angegeben.) Sobald der von RTIMOUT angegebene Zeitraum abgelaufen ist, gibt CICS eine Antwort TIMEOUT an die Anwendung zurück. Der Wert Null für RTIMOUT bedeutet, dass die Anwendung unbe-

grenzt warten kann. Die Standardeinstellung für RTIMOUT bei Definitionen von Transaktionsprofilen ist Null; daher muss diese Einstellung für Anwendungen, die HTTP-Clientanforderungen ausgeben, unbedingt überprüft und geändert werden.

Die Anforderung kann ebenfalls ein Zeitlimit überschreiten, wenn eine Nachricht an den Server gesendet wird. In diesem Fall gilt das Zeitlimitintervall für Deadlocks, das im Attribut DTIMOUT der Transaktionsdefinition angegeben ist, und CICS gibt eine Antwort TIMEDOUT an die Anwendung zurück.

Der Befehl CONVERSE unterstützt das Chunked Transfer Coding für die Anwendung nicht, da dies eine Folge von Sendeaktionen erforderlich macht und der Befehl WEB CONVERSE nur eine einzige Sendeaktion bereitstellt. Falls Sie eine in Blöcke aufgeteilte Nachricht senden wollen, verwenden Sie zum Senden den Befehl WEB SEND und zum Empfangen den Befehl WEB RECEIVE. Wenn der Server eine in Blöcke aufgeteilte Antwort sendet, kann diese Antwort mit dem Befehl WEB CONVERSE empfangen werden.

Der Befehl WEB CONVERSE kann nicht verwendet werden, nachdem die Verbindung zum Server geschlossen wurde. Falls Sie testen müssen, ob der Server das Beenden der Verbindung angefordert hat, suchen Sie mit dem Befehl WEB READ HTTPHEADER in der letzten Nachricht vom Server nach dem Header 'Connection: close'.

Der Befehl WEB CONVERSE führt eine einzige Sendeaktion und eine einzige Empfangsaktion aus; er kann anstelle eines Befehls WEB SEND und eines Befehls WEB RECEIVE verwendet werden. Sie können Befehle WEB SEND und WEB RECEIVE sowie Befehle WEB CONVERSE im Zusammenhang mit derselben Verbindung (also mit demselben Sitzungstoken) verwenden. Findet jedoch ein Pipelining von Anforderungen statt (also das Senden einer Folge von Anforderungen ohne Warten auf eine Antwort), dürfen Sie nach einem Befehl WEB SEND keinen Befehl WEB CONVERSE verwenden. CICS überprüft zur Programmlaufzeit, ob es für jeden Befehl WEB SEND einen nachfolgenden Befehl WEB RECEIVE gibt, bevor ein Befehl WEB CONVERSE ausgegeben wird. Falls Sie beispielsweise den Befehl WEB SEND drei Mal zum Ausgeben einer Pipelinefolge von Anforderungen verwenden, müssen Sie den Befehl WEB RECEIVE drei Mal verwenden, um die Antworten für diese Anforderungen zu empfangen, bevor Sie den Befehl WEB CONVERSE verwenden können.

HTTP-Header werden normal unter Verwendung des 64-Bit-Speichers (oberhalb der 2-GB-Grenze) empfangen, verarbeitet, aufbewahrt und gesendet. Wenn der Hauptteil der HTTP-Clientanforderung in einem Container empfangen oder von dort aus gesendet wird, werden die Containerdaten unter Verwendung des 64-Bit-Speichers empfangen, verarbeitet und gesendet. Wenn der HTTP-Hauptteil in einem Anwendungspuffer empfangen wird und CICS diese Daten aufbewahren muss, bevor sie an die Anwendung zurückgegeben werden können, bewahrt CICS die Daten in 64-Bit-Puffern auf. Wenn der HTTP-Hauptteil aus einem Anwendungspuffer gesendet wird und CICS die Daten konvertieren muss, werden die Daten in einem 64-Bit-Puffer konvertiert und von dort aus gesendet. Beim Senden eines Dokuments wird das Dokument jedoch im 31-Bit-Speicher (oberhalb der 16-MB-Grenze) verarbeitet.

Optionen für das Senden von HTTP-Clientanforderungen

ACTION(cvda)

Mit dieser Option wird angegeben, wie die Nachricht gesendet werden soll. Der CVDA-Wert lautet wie folgt:

EXPECT

Weist CICS an, einen Header 'Expect' zusammen mit der Anforderungszeile und den Headern für die Anforderung zu senden und eine Antwort '100-Continue' abzuwarten, bevor der Nachrichtenhauptteil an den Server gesendet wird. Falls eine andere Antwort als '100-Continue' empfangen wird, informiert CICS das Anwendungsprogramm und bricht den Sendevorgang ab. Falls nach einer Wartezeit keine Antwort empfangen wird, sendet CICS den Nachrichtenhauptteil trotzdem.

Der Header 'Expect' wird von Servern mit einer älteren Version als HTTP 1.1 nicht unterstützt. Falls CICS die HTTP-Version des Servers noch nicht kennt, gibt CICS vor dem Senden Ihrer Anforderung eine zusätzliche Anforderung aus, um die HTTP-Version des Servers zu ermitteln. Falls der Header 'Expect' nicht geeignet ist, sendet CICS Ihre Anforderung ohne ihn.

Diese Option darf nur verwendet werden, wenn Ihre Anforderung einen Nachrichtenhauptteil besitzt.

AUTHENTICATE(*cvda*)

Mit dieser Option können Sie Details (Berechtigungsnachweise) für die Benutzerauthentifizierung angeben, um den Zugriff auf eingeschränkte Daten zu steuern. Die CVDA-Werte lauten wie folgt:

NONE

Gibt an, dass keine Einschränkungen für den Zugriff auf diese Daten bestehen und daher keine Berechtigungsnachweise erforderlich sind. Dies ist der Standardwert für die Option AUTHENTICATE.

BASICAUTH

Gibt an, dass bei dieser Sitzung Berechtigungsnachweise für die HTTP-Basisauthentifizierung erforderlich sind. Diese Details können im Befehl selbst oder mithilfe des globalen Benutzerexits XWBAUTH angegeben werden.

CHANNEL(*datenwert*)

Gibt den Namen des Kanals an, zu dem der Container gehört. Der Name des Kanals kann aus bis zu 16 alphanumerischen Zeichen inklusive der entsprechenden Interpunktion bestehen. Führende und eingebettete Leerzeichen sind nicht zulässig. Falls der Name weniger als 16 Zeichen umfasst, wird er mit nachgestellten Leerzeichen aufgefüllt. Sie können den Kanalnamen DFHTRANSACTION angeben, um den Transaktionskanal zu verwenden.

Falls die Option CONTAINER angegeben ist, ist die Option CHANNEL optional.

Ist die Option CHANNEL nicht angegeben, geht CICS vom aktuellen Kanal aus.

CHARACTERSET(*datenwert*)

Gibt einen Zeichensatz an, in den CICS den Entitätshauptteil des mit dem Befehl gesendeten Elements vor dem Sendevorgang konvertiert. Der Name des Zeichensatzes kann aus bis zu 40 alphanumerischen Zeichen inklusive der entsprechenden Interpunktion bestehen. CICS unterstützt nicht alle von IANA genannten Zeichensätze. Im Abschnitt [HTML-codierte Zeichensätze](#) sind die IANA-Zeichensätze aufgelistet, die durch CICS für die Codepagekonvertierung unterstützt werden.

Damit eine Codepagekonvertierung des Entitätshauptteils möglich ist, muss die Option CLIENTCONV auf CLICONVERT festgelegt sein bzw. den Standardwert verwenden können. Falls die Option NOCLICONVERT angegeben ist, findet keine Codepagekonvertierung statt. Wenn eine Konvertierung angefordert wird und die Option CHARACTERSET nicht angegeben ist, wird ISO-8859-1 als Standardzeichensatz verwendet.

CLOSESTATUS(*cvda*)

Gibt an, ob ein Verbindungsheader mit der Verbindungsoption 'close' (Connection: close) in die Anforderung einbezogen werden soll oder nicht. Standardmäßig wird der Header nicht einbezogen. Die CVDA-Werte lauten wie folgt:

CLOSE

Veranlasst CICS, für diese Anforderung einen Header 'Connection: close' zu schreiben. Der Header benachrichtigt den Server, dass die Verbindung geschlossen werden soll, nachdem der Server seine Antwort auf die Anforderung gesendet hat. (Bei einem Server mit Version HTTP/1.0, erzielt CICS dieselbe Wirkung, indem der Header 'Connection: Keep-Alive' weggelassen wird.) Geben Sie diese Option nicht an, falls Sie das Verbindungspooling in der URIMAP-Ressource für diese Verbindung implementiert haben, weil eine geschlossene Verbindung nicht für die Wiederverwendung in den Pool gestellt werden kann. Geben Sie diese Option ausschließlich dann an, wenn Sie die letzte Anforderung an den Server ausgeben und das Verbindungspooling nicht verwendet wird.

NOCLOSE

Bedeutet, dass der Header 'Connection: close' nicht für diese Anforderung verwendet wird. Falls für den Server die Version HTTP/1.0 erkannt wird, sendet CICS einen Verbindungsheader mit der Verbindungsoption 'Keep-Alive' (Connection: Keep-Alive), um den Server zu benachrichtigen, dass eine persistente Verbindung gewünscht wird.

CONTAINER(*datenwert*)

Gibt den Namen des Containers an, in dem der HTTP-Hauptteil aufbewahrt wird, bevor er an den Server gesendet wird. Der Name des Containers kann aus bis zu 16 alphanumerischen Zeichen inklusive der entsprechenden Interpunktion bestehen. Führende und eingebettete Leerzeichen sind nicht zu-

lässig. Falls der Name weniger als 16 Zeichen umfasst, wird er mit nachgestellten Leerzeichen aufgefüllt.

Bei Angabe der Option CONTAINER speichert CICS den HTTP-Hauptteil im 64-Bit-Speicher und sendet ihn von dort aus.

DOCSTATUS(*cvda*)

Gibt an, ob das Dokument während der Verarbeitung des Befehls WEB CONVERSE gelöscht werden soll oder nicht. Die CVDA-Werte lauten wie folgt:

DOCDELETE

CICS löscht das Dokument, nachdem der Dokumentinhalt für den Sendevorgang gespeichert wurde. Der dem Dokument zugeordnete Speicher wird sofort freigegeben. Falls Sie nachfolgende Anforderungen für das Dokument ausgeben, generieren diese eine Antwort TOKENERR.

NODOCDELETE

CICS löscht das Dokument während der Verarbeitung des Befehls **WEB CONVERSE** nicht. Dies ist der Standardwert für die Option DOCSTATUS.

DOCTOKEN(*datenwert*)

Gibt das 16 Byte umfassende binäre Token eines Dokuments an, das als Nachrichtenhauptteil gesendet werden soll. Das Dokument muss mit der CICS-Dokumentschnittstelle unter Verwendung der EXEC CICS-Befehle **DOCUMENT CREATE**, **INSERT** und **SET** erstellt worden sein. Als alternatives Verfahren für die Erstellung eines Nachrichtenhauptteils steht die Option FROM zur Verfügung.

Bei Angabe der Option DOCTOKEN wird das Dokument im 31-Bit-Speicher verarbeitet.

FROM(*datenbereich*)

Gibt einen Puffer von Daten an, in dem sich der Nachrichtenhauptteil befindet. Der Nachrichtenhauptteil wird durch das Anwendungsprogramm erstellt. Wenn Sie die Option FROM angeben, müssen Sie mit der Option FROMLENGTH die Länge des Datenpuffers angeben. Als alternatives Verfahren für die Erstellung eines Nachrichtenhauptteils steht die Option DOCTOKEN zur Verfügung.

Für die Größe des Datenbereichs gibt es keinen Maximalwert; aufgrund von Speicheraspekten ist seine Größe in der Praxis jedoch begrenzt. Der Abschnitt Entitätshauptteil für eine HTTP-Nachricht erstellen enthält weitere Informationen zu Speicheraspekten.

Wenn die Option FROM angegeben ist und CICS den HTTP-Hauptteil vor dem Senden kopieren oder konvertieren muss, wird der Hauptteil im 64-Bit-Speicher kopiert oder konvertiert und von dort aus gesendet.

FROMLENGTH(*datenbereich*)

Gibt als Vollwort-Binärwert die Länge des Datenpuffers an, der in der Option FROM (für den Nachrichtenhauptteil) angegeben ist. Dieser Wert muss unbedingt korrekt angegeben werden, weil eine falsche Datenlänge für den Empfänger der Nachricht zu Problemen führen kann.

MEDIATYPE(*datenbereich*)

Gibt den Dateninhalt eines bereitgestellten Nachrichtenhauptteils an, z. B. text/xml. Als Wert für die Option MEDIATYPE müssen Sie einen 56 Byte umfassenden Bereich angeben. Der Medientyp kann bis zu 56 alphanumerische Zeichen lang sein, inklusive der entsprechenden Interpunktion, jedoch ohne Leerzeichen. Weitere Informationen zu Medientypen finden Sie unter IANA-Datenträgertypen und -Zeichensätze. CICS überprüft, ob das Format des Medientyps richtig ist, nimmt jedoch keine Überprüfung für die Gültigkeit des Medientyps in Bezug auf den Dateninhalt vor. CICS erstellt anhand dieser Informationen den Header 'Content-Type' für die Nachricht.

Bei Anforderungen, die einen Hauptteil erfordern, müssen Sie die Option MEDIATYPE angeben. Es gibt keinen Standardwert. Falls der erforderliche Header 'Content-Type' jedoch Leerzeichen oder mehr als 56 Zeichen enthalten muss, kann die Anwendung ihn mit dem Befehl WEB WRITE HTTPHEADER bereitstellen. Geben Sie in diesem Fall die Option MEDIATYPE nicht an.

Anhand des bereitgestellten Medientyps wird unter den folgenden Umständen ermittelt, ob eine Codepagekonvertierung erforderlich ist:

- Sie senden mit der Option FROM eine Nachricht aus einem Puffer und die Optionen CLIENTCONV und CHARACTERSET sind nicht angegeben.

- Sie senden mit der Option DOCTOKEN eine Nachricht aus einem Dokument und die Optionen CLIENTCONV und CHARACTERSET sind nicht angegeben.
- Sie senden mit der Option CONTAINER eine Nachricht aus einem benannten Container und entweder ist die Option CLICONVERT angegeben oder die Optionen CLIENTCONV und CHARACTERSET sind nicht angegeben.

Falls Text als Medientyp angegeben ist, wird die Nachricht konvertiert. Ist nicht Text als Medientyp angegeben, wird die Nachricht nicht konvertiert.

Die Option MEDIATYPE wird für die Sende- und auch die Empfangsfunktion des Befehls WEB CONVERSE verwendet. Falls die Option mit einem Wert angegeben ist, wird der Wert zum Erstellen des Headers 'Content-Type' in der Anforderung verwendet und dasselbe Feld wird zum Empfangen des Medientyps für die Antwort verwendet, die durch den Server zurückgegeben wird. Falls die Option ohne einen Wert angegeben ist, wird sie lediglich verwendet, um den Medientyp der Antwort zu empfangen.

METHOD(*cvda*)

Gibt die HTTP-Methode für die Anforderung an.

Von diesem Befehl werden die Methoden GET, HEAD, PATCH, POST, PUT, TRACE, OPTIONS und DELETE unterstützt. Manche HTTP-Server, insbesondere Server mit der Version HTTP/1.0, implementieren jedoch möglicherweise nicht alle dieser Methoden.

Der Abschnitt Entitätshauptteil für eine HTTP-Nachricht erstellen enthält weitere Informationen zur korrekten Verwendung von Methoden, inklusive der HTTP-Versionen, die für die einzelnen Methoden gelten.

CICS blockiert das Senden eines Nachrichtenhauptteils für Methoden, bei denen er ungeeignet ist, und macht ihn für Methoden erforderlich, bei denen er geeignet ist.

Die CVDA-Werte lauten wie folgt:

GET

Fordert eine Ressource vom Server an. Ein Anforderungshauptteil ist nicht zulässig.

HEAD

Fordert für eine Ressource die HTTP-Header, jedoch nicht den Antworthauptteil an. Ein Anforderungshauptteil ist nicht zulässig.

PATCH

Fordert an, dass eine Gruppe von Änderungen auf die durch den Anforderungs-URI identifizierte Ressource angewendet wird. Ein Anforderungshauptteil ist erforderlich.

POST

Sendet Daten an einen Server. Ein Anforderungshauptteil ist erforderlich.

PUT

Erstellt oder ändert eine Ressource auf dem Server. Ein Anforderungshauptteil ist erforderlich.

TRACE

Erstellt einen Trace für die Route Ihrer Anforderung zum Server. Ein Anforderungshauptteil ist nicht zulässig.

OPTIONS

Ruft Informationen zum Server ab. Ein Anforderungshauptteil ist zulässig, aber es gibt keinen definierten Zweck für den Hauptteil. Falls Sie einen Anforderungshauptteil verwenden, müssen Sie einen Medientyp angeben.

DELETE

Löscht eine Ressource auf dem Server. Ein Anforderungshauptteil ist nicht zulässig.

PASSWORD(*datenwert*)

Gibt das Kennwort an, das der Benutzer-ID oder dem Anmeldenamen zugeordnet ist, die/der auf diese Daten zugreifen darf. Die Option PASSWORD ist nur erforderlich, wenn die Option USERNAME verwendet wird. Falls das angegebene Kennwort länger als 8 Zeichen ist, wird es als Kennwortphrase behandelt, wenn es an z/OS-Systeme gesendet wird.

PASSWORDLEN(datenwert)

Gibt als Vollwort-Binärzahlvariable die Länge des für die Option PASSWORD angegebenen Puffers an.

PATH(datenbereich)

Gibt die Pfadinformationen für die spezielle Ressource im Server an, auf die die Anwendung Zugriff benötigt.

Falls mit der Option URIMAP im Befehl WEB OPEN für diese Verbindung eine vorhandene URIMAP-Definition angegeben wurde, ist der in der URIMAP-Definition festgelegte Pfad der Standardpfad für den Befehl WEB SEND. Falls Sie unter diesen Umständen keine Pfadinformationen im Befehl WEB SEND angeben, wird der Pfad aus der URIMAP-Definition verwendet. Falls Sie einen anderen Pfad angeben, als in der URIMAP-Definition festgelegt ist, überschreibt dieser Wert den Pfad aus der URIMAP-Definition.

Für den Fall, dass die Option URIMAP im Befehl WEB OPEN nicht verwendet wurde, gibt es keinen Standardpfad und Sie müssen Pfadinformationen angeben. Pfadinformationen können mit dem Befehl WEB PARSE URL aus einer bekannten URL extrahiert werden.

Als Alternative zur Angabe der Pfadinformationen mit der Option PATH können Sie auch mit der Option URIMAP des Befehls WEB CONVERSE eine URIMAP-Definition angeben, aus der die Pfadinformationen direkt entnommen werden.

PATHLENGTH(datenwert)

Gibt als Vollwort-Binärwert die Länge des Pfades an. Falls Sie mit der Option PATH Pfadinformationen bereitstellen, müssen Sie die Option PATHLENGTH angeben. Die Informationen zur Pfadlänge werden zurückgegeben, wenn Sie eine URL mit dem Befehl WEB PARSE URL analysieren.

QUERYSTRING(datenbereich)

Gibt eine Abfragezeichenfolge an, die dem Server als Teil der Anforderung übergeben werden soll. Die Angabe eines Fragezeichens (?) am Beginn der Abfragezeichenfolge ist nicht zwingend erforderlich; falls Sie das Fragezeichen nicht angeben, stellt CICS es beim Erstellen der Anforderung automatisch für Sie bereit. Wenn Sie Zeichen mit Escapezeichen in die Abfragezeichenfolge einbeziehen, übergibt CICS die Zeichen in ihrem Format nach Anwendung der Escapezeichen an den Server.

QUERYSTRLEN(datenwert)

Gibt als Vollwort-Binärwert die Länge der in der Option QUERYSTRING bereitgestellten Abfragezeichenfolge an.

SESTOKEN(datenwert)

Gibt den 8 Byte umfassenden Binärwert für das Sitzungstoken an, das diese Verbindung zwischen CICS und einem Server eindeutig kennzeichnet. Dieser Wert wird durch einen Befehl WEB OPEN für CICS als HTTP-Client zurückgegeben. Die Verwendung des Sitzungstokens ist im Abschnitt [Sitzungstokens](#) erläutert.

URIMAP(datenwert)

Gibt den Namen (bis zu 8 Zeichen in Groß-/Kleinschreibung) einer URIMAP-Definition an, die Pfadinformationen für die jeweilige Ressource im Server bereitstellt, auf die die Anwendung Zugriff benötigt. Die URIMAP-Definition muss für CICS als HTTP-Client (mit Angabe von USAGE(CLIENT)) bestimmt sein. Ihr Attribut HOST muss mit dem Attribut HOST der URIMAP-Definition, die im Befehl WEB OPEN für diese Verbindung angegeben wurde, oder mit dem Hostnamen identisch sein, der in der Option HOST des Befehls WEB OPEN für diese Verbindung angegeben wurde. Eine im Befehl WEB CONVERSE angegebene URIMAP-Definition gilt nur für diese Anforderung.

Geben Sie die Option PATH oder PATHLENGTH nicht an, falls die Option URIMAP angegeben wird.

USERNAME(datenwert)

Gibt die Benutzer-ID oder den Anmeldenamen an, die/der auf diese Daten zugreifen darf. Bei Angabe der Option USERNAME müssen Sie auch die Option PASSWORD verwenden.

USERNAMELEN(datenwert)

Gibt als Vollwort-Binärzahlvariable die Länge des für die Option USERNAME angegebenen Puffers an.

Optionen für das Empfangen von Serverantworten

BODYCHARSET(datenbereich)

Gibt den Zeichensatz des HTTP-Antworthauptteils an.

Der Name des Zeichensatzes kann aus bis zu 40 alphanumerischen Zeichen inklusive der entsprechenden Interpunktion bestehen.

Wenn der HTTP-Hauptteil in einem Anwendungspuffer empfangen wird, wird der Zeichensatz wie folgt zurückgegeben:

- Falls die Option INTO oder SET angegeben ist und der HTTP-Hauptteil konvertiert wird, gibt CICS den Zeichensatz des HTTP-Hauptteils vor der Konvertierung zurück.
- Falls die Option INTO oder SET angegeben ist und der HTTP-Hauptteil nicht konvertiert wird, gibt CICS den Zeichensatz zurück, der im Header 'Content-Type' angegeben ist. Wenn keine Zeichensatzinformationen verfügbar sind, werden Leerzeichen zurückgegeben.

Wenn der HTTP-Hauptteil in einem benannten Container empfangen wird, wird der Zeichensatz wie folgt zurückgegeben:

- Falls der Container den Datentyp CHAR besitzt, gibt CICS den Zeichensatz der codierten Daten zurück.
- Falls der Container den Datentyp BIT besitzt, gibt CICS Leerzeichen zurück.

Wenn der zurückgegebene Wert mehr als 40 Byte umfasst, werden die Daten abgeschnitten. Umfasst der zurückgegebene Wert weniger als 40 Byte, werden die Daten rechts mit Leerzeichen aufgefüllt.

INTO(datenbereich)

Gibt den Puffer an, der die empfangenen Daten enthalten soll.

Falls die HTTP-Anforderung einen Header 'Content-Length' enthält, bewahrt CICS den HTTP-Hauptteil im internen 64-Bit-Speicher auf; andernfalls bewahrt CICS den Hauptteil im internen 31-Bit-Speicher auf. Der Hauptteil wird aus diesem internen Speicher in den Anwendungspuffer kopiert, der in der Option **INTO** angegeben ist.

MAXLENGTH(datenwert)

Gibt als Vollwort-Binärwert die maximale Datenmenge an, die CICS an die Anwendung übergeben soll. Die Option MAXLENGTH gilt unabhängig davon, ob die Option INTO oder SET für den Empfang von Daten angegeben ist. Falls die Daten unter Verwendung von Chunked Transfer Coding gesendet wurden, assembliert CICS die Blöcke in einer einzigen Nachricht, bevor diese an die Anwendung übergeben wird. Die Option MAXLENGTH gilt daher für die Gesamtlänge der in Blöcke aufgeteilten Nachricht und nicht für jeden einzelnen Block. Die Messung der Daten findet nach einer gegebenenfalls erfolgten Codepagekonvertierung statt.

Falls die Länge der Daten den angegebenen Wert überschreitet und die Option NOTRUNCATE nicht angegeben ist, werden die Daten auf diesen Wert abgeschnitten und die übrigen Daten werden gelöscht.

Falls die Länge der Daten den angegebenen Wert überschreitet und die Option NOTRUNCATE angegeben ist, bewahrt CICS die verbleibenden Daten auf und verwendet sie zur Erfüllung nachfolgender Befehle RECEIVE.

MEDIATYPE(datenbereich)

Gibt einen 56 Zeichen umfassenden Datenbereich für den Empfang des Medientyps für den Hauptteil (also den Typ des Dateninhalts) an, z. B. `text/xml`. Weitere Informationen zu Medientypen finden Sie unter [IANA-Datenträgertypen](#) und [-Zeichensätze](#).

Die Option MEDIATYPE wird für die Sende- und auch die Empfangsfunktion des Befehls WEB CONVERSE verwendet. Falls die Option mit einem Wert angegeben ist, wird der Wert zum Erstellen des Headers 'Content-Type' in der Anforderung verwendet und dasselbe Feld wird zum Empfangen des Medientyps für die Antwort verwendet, die durch den Server zurückgegeben wird. Falls die Option ohne einen Wert angegeben ist, wird sie lediglich verwendet, um den Medientyp der Antwort zu empfangen.

NOTRUNCATE

Gibt an, dass bei einer Überschreitung der in der Option MAXLENGTH angeforderten Länge durch die verfügbaren Daten die übrigen Daten nicht sofort gelöscht, sondern zum Abruf durch nachfolgende Befehle RECEIVE aufbewahrt werden sollen. (Falls keine weiteren Befehle RECEIVE ausgegeben werden, werden die Daten bei der Beendigung der Transaktion gelöscht.)

Ein einzelner Befehl RECEIVE mit der Option SET und ohne die Option MAXLENGTH empfängt alle verbleibenden Daten ungeachtet ihrer Länge. Alternativ können Sie eine Folge von Befehlen RECEIVE mit der Option NOTRUNCATE verwenden, um die verbleibenden Daten in geeigneten Blöcken zu empfangen. Geben Sie so lang Befehle RECEIVE aus, bis Sie keine Antwort LENGERR mehr empfangen. Denken Sie daran, dass der Empfang einer kürzeren Länge, als in der Option MAXLENGTH angegeben ist, nicht zwangsläufig ein Hinweis auf das Ende der Daten ist. Diese Situation könnte vorliegen, wenn CICS die Rückgabe eines abgeschnittenen Zeichens am Ende der Daten vermeiden muss.

SET(zeigerverweis)

Gibt den Zeigerverweis an, der auf die Adresse der empfangenen Daten gesetzt werden soll. Der Zeigerverweis ist gültig, bis der nächste Befehl RECEIVE mit demselben Sitzungstoken ausgegeben wird, oder bis zum Ende der Task.

Falls die HTTP-Anforderung einen Header 'Content-Length' enthält, bewahrt CICS den HTTP-Hauptteil im internen 64-Bit-Speicher auf; andernfalls bewahrt CICS den Hauptteil im internen 31-Bit-Speicher auf. Der Hauptteil wird aus diesem internen Speicher in den Speicher kopiert, der für die Anwendung zugänglich ist, und der Zeigerverweis wird auf diese Kopie gesetzt.

STATUSCODE(datenbereich)

Gibt einen Datenbereich für den Empfang des HTTP-Statuscodes an, der durch den Server gesendet wird. Der Code ist ein Halbwort-Binärwert. Beispiele sind 200 (normal) oder 404 (nicht gefunden). Der Empfang des Statuscodes ist optional. Unter den folgenden Umständen sollten Sie jedoch den Statuscode in jedem Fall empfangen und überprüfen:

- Sie wollen entweder jetzt oder während einer künftigen Verbindung eine identische Anforderung an den Server ausgeben.
- Sie wollen unter Verwendung dieser Verbindung weitere Anforderungen an den Server ausgeben.
- Ihre Anwendung führt eine weitere Verarbeitung aus, die von den Informationen abhängig ist, die Sie in der Antwort empfangen..

Der Abschnitt [HTTP-Statuscodereferenz für CICS-Webunterstützung](#) enthält grundlegende Anleitungen für entsprechende Aktionen, die eine Anwendung als Reaktion auf die Statuscodes für HTTP/1.1 ausführen muss.

STATUSLEN(datenwert)

Gibt als Vollwort-Binärwert die Länge des Datenbereichs an, der einen vom Server zurückgegebenen Text für die Beschreibung des Statuscodes (Option STATUSTEXT) empfangen soll. Der Text wird als 'Ursachenphrase' bezeichnet. Die meisten für HTTP empfohlenen Ursachenphrasen sind kurz, aber für den Fall, dass der Server den empfohlene Ursachencode durch detailliertere Informationen ersetzt, wird hier für den Datenbereich eine Länge von 256 empfohlen.

STATUSTEXT(datenbereich)

Gibt einen Datenbereich an, der einen vom Server zurückgegebenen Text für die Beschreibung des Statuscodes empfangen soll. Der Text wird als 'Ursachenphrase' bezeichnet. Beispiele sind 'OK' (mit einem Statuscode 200) oder 'Bad Request' (mit einem Statuscode 400). Die zulässige Länge für den Text wird in der Option STATUSLEN angegeben.

TOCHANNEL(datenwert)

Gibt den Namen des Kanals an, zu dem der Container gehört. Der Name des Kanals kann aus bis zu 16 alphanumerischen Zeichen inklusive der entsprechenden Interpunktion bestehen. Zulässige Zeichen: A-Z a-z 0-9 \$ @ # / % & ? ! : | " = , ; < > . - und _ . Führende und eingebettete Leerzeichen sind nicht zulässig. Falls der Name weniger als 16 Zeichen umfasst, wird er mit nachgestellten Leerzeichen aufgefüllt. Falls der Kanal nicht vorhanden ist, wird er erstellt. Dieser neue Kanal bleibt gültig, bis sich die Verbindungsebene ändert. Informationen zum Geltungsbereich von Kanälen finden Sie unter [Geltungsbereich eines Kanals](#).

Wenn Sie Ihre Kanäle zwischen CICS-Regionen verlagern wollen, denken Sie daran, dass Sie ausschließlich alphanumerische Zeichen (A-Z 0-9 & : = , ; < > . - _) verwenden; dies stellt sicher, dass die Zeichen von allen EBCDIC-Codepages identisch dargestellt werden.

Sie können den Kanalnamen DFHTRANSACTION angeben, um einen Transaktionskanal zu verwenden. Ein Transaktionskanal verliert nicht seine Gültigkeit, wenn sich die Verbindungsebene ändert; der Zugriff auf diesen Kanal ist in der Transaktion immer möglich. Weitere Informationen finden Sie unter Kanäle und Container.

Falls die Option TOCHANNEL nicht angegeben ist, geht CICS vom aktuellen Kanal aus.

Bei Angabe der Option TOCONTAINER bewahrt CICS den empfangenen Hauptteil der HTTP-Anforderung im 64-Bit-Speicher auf, bis die Anwendung den HTTP-Hauptteil mit einem API-Befehl CONTAINER abruft.

TOCONTAINER(datenwert)

Gibt den Namen des Containers an, in den die Daten gestellt werden. Der Name des Containers kann aus bis zu 16 alphanumerischen Zeichen inklusive der entsprechenden Interpunktion bestehen. Zulässige Zeichen: A-Z a-z 0-9 \$ @ # / % & ? ! : | " = , ; < > . - und _ . Führende und eingebettete Leerzeichen sind nicht zulässig. Falls der Name weniger als 16 Zeichen umfasst, wird er mit nachgestellten Leerzeichen aufgefüllt.

Wenn Sie Ihre Container zwischen CICS-Regionen verlagern wollen, denken Sie daran, dass Sie ausschließlich alphanumerische Zeichen (A-Z 0-9 & : = , ; < > . - _) verwenden; dies stellt sicher, dass die Zeichen von allen EBCDIC-Codepages identisch dargestellt werden.

Verwenden Sie nur dann Containernamen, die mit 'DFH' beginnen, wenn Sie hierzu von CICS aufgefordert werden.

Bei Angabe der Option TOCONTAINER bewahrt CICS den empfangenen Hauptteil der HTTP-Anforderung im 64-Bit-Speicher auf, bis die Anwendung den HTTP-Hauptteil mit einem API-Befehl CONTAINER abruft.

TOLENGTH(datenbereich)

Gibt eine Vollwort-Binärzahlvariable an, die auf die Menge der Daten gesetzt wird, die CICS an die Anwendung zurückgegeben hat. Dieser Wert kann - insbesondere bei Einbeziehung eines Doppelbyte- oder Mehrbytezeichensatzes - etwas niedriger als der mit der Option MAXLENGTH festgelegte Grenzwert sein, weil CICS am Ende der Daten kein abgeschnittenes Zeichen zurückgibt.

- Falls die Option NOTRUNCATE nicht angegeben ist, werden alle weiteren Daten in der Nachricht gelöscht. Falls weitere Daten vorhanden sind, wird eine Antwort LENGERR mit dem RESP2-Wert 57 zurückgegeben.
- Falls die Option NOTRUNCATE angegeben ist, werden alle zusätzlichen Daten aufbewahrt. Falls zusätzliche Daten vorhanden sind, wird eine Antwort LENGERR mit dem RESP2-Wert 36 zurückgegeben. Die Beschreibung für die Option NOTRUNCATE enthält Anweisungen für einen solchen Fall.

Diese Option ist die funktionale Entsprechung der Option LENGTH im Befehl WEB RECEIVE.

Geben Sie die Option TOLENGTH nicht an, falls Sie die Option TOCONTAINER verwenden.

Optionen für das Konvertieren von gesendeten und empfangenen Elementen

CLIENTCONV(cvda)

Gibt an, ob CICS den Entitätshauptteil der HTTP-Anforderung vor dem Senden konvertiert und den Entitätshauptteil der Serverantwort konvertiert oder nicht. Standardmäßig wird der Entitätshauptteil sowohl beim Senden der Anforderung als auch beim Empfangen der Antwort konvertiert (CLICONVERT).

Falls Sie Daten in einem benannten Container empfangen (angegeben durch die Option TOCONTAINER), findet keine Konvertierung statt.

- Für den Anforderungshauptteil können Sie mit der Option CHARACTERSET in diesem Befehl einen Zeichensatz angeben, der für den Server geeignet ist. Falls eine Konvertierung angefordert wird

(oder standardmäßig stattfindet), Sie jedoch keinen Zeichensatz angeben, konvertiert CICS den Entitätshauptteil standardmäßig in den Zeichensatz ISO-8859-1.

- Für den Antworthauptteil müssen Sie den vom Server verwendeten Zeichensatz nicht angeben. CICS ermittelt diesen Wert durch eine Untersuchung des Headers 'Content-Type' in der Nachricht. Falls diese Informationen nicht im Header bereitgestellt werden oder der benannte Zeichensatz von CICS für die Codepagekonvertierung nicht unterstützt wird, wird der Zeichensatz ISO-8859-1 verwendet.
- Für die Anwendungscodepage wird die Standardcodepage der lokalen CICS-Region (gemäß der Angabe im Systeminitialisierungsparameter LOCALCCSID) oder eine alternative EBCDIC-Codepage verwendet, die Sie im Befehl WEB OPEN angegeben haben.

Die CVDA-Werte lauten wie folgt:

CLICONVERT

CICS konvertiert den Entitätshauptteil der Anforderung in den Zeichensatz, den Sie für den Server angeben, und den Entitätshauptteil der Antwort in eine Codepage, die für die Anwendung geeignet ist.

Falls die Option TOCONTAINER angegeben ist, findet beim Empfang der HTTP-Antwort keine Konvertierung statt. Stattdessen bestimmt der Medientyp im HTTP-Antwortheader, ob der HTTP-Hauptteil in einem Container mit dem Datentyp BIT oder CHAR gespeichert wird. Falls der Medientyp ein Textmedientyp ist, wird der Hauptteil in einem Container mit dem Datentyp CHAR gespeichert. Ist der Medientyp ein nicht für Text bestimmter Medientyp, wird der Hauptteil in einem Container mit dem Datentyp BIT gespeichert. Falls die HTTP-Antwort keine Informationen zum Medientyp enthält, wird standardmäßig vom Textmedientyp ausgegangen.

NOINCONVERT

CICS konvertiert den Entitätshauptteil der Anforderung in den Zeichensatz, den Sie für den Server angeben. CICS konvertiert jedoch nicht den Entitätshauptteil der Antwort, der an die Anwendung in dem vom Server verwendeten Zeichensatz übergeben wird.

Falls die Option TOCONTAINER angegeben ist, findet beim Empfang der HTTP-Antwort keine Konvertierung statt. Stattdessen bestimmt der Medientyp im HTTP-Antwortheader, ob der HTTP-Hauptteil in einem Container mit dem Datentyp BIT oder CHAR gespeichert wird. Falls der Medientyp ein Textmedientyp ist, wird der Hauptteil in einem Container mit dem Datentyp CHAR gespeichert. Ist der Medientyp ein nicht für Text bestimmter Medientyp, wird der Hauptteil in einem Container mit dem Datentyp BIT gespeichert. Falls die HTTP-Antwort keine Informationen zum Medientyp enthält, wird standardmäßig vom Textmedientyp ausgegangen.

NOOUTCONVERT

CICS konvertiert den Entitätshauptteil der Anforderung nicht; dieser wird in dem von der Anwendung verwendeten Zeichensatz an den Server gesendet. CICS konvertiert jedoch den Entitätshauptteil der Antwort in eine Codepage, die für die Anwendung geeignet ist.

Falls die Option TOCONTAINER angegeben ist, findet beim Empfang der HTTP-Antwort keine Konvertierung statt. Stattdessen bestimmt der Medientyp im HTTP-Antwortheader, ob der HTTP-Hauptteil in einem Container mit dem Datentyp BIT oder CHAR gespeichert wird. Falls der Medientyp ein Textmedientyp ist, wird der Hauptteil in einem Container mit dem Datentyp CHAR gespeichert. Ist der Medientyp ein nicht für Text bestimmter Medientyp, wird der Hauptteil in einem Container mit dem Datentyp BIT gespeichert. Falls die HTTP-Antwort keine Informationen zum Medientyp enthält, wird standardmäßig vom Textmedientyp ausgegangen.

NOCLICONVERT

CICS konvertiert den Entitätshauptteil der Anforderung nicht; dieser wird in dem von der Anwendung verwendeten Zeichensatz an den Server gesendet. CICS konvertiert nicht den Entitätshauptteil der Antwort, der an die Anwendung in dem vom Server verwendeten Zeichensatz übergeben wird.

Falls die Option TOCONTAINER angegeben ist, findet beim Empfang der HTTP-Antwort keine Konvertierung statt. Stattdessen bestimmt der Medientyp im HTTP-Antwortheader, ob der HTTP-Hauptteil in einem Container mit dem Datentyp BIT oder CHAR gespeichert wird. Falls der Medi-

entyp ein Textmedientyp ist, wird der Hauptteil in einem Container mit dem Datentyp CHAR gespeichert. Ist der Medientyp ein nicht für Text bestimmter Medientyp, wird der Hauptteil in einem Container mit dem Datentyp BIT gespeichert. Falls die HTTP-Antwort keine Informationen zum Medientyp enthält, wird standardmäßig vom Textmedientyp ausgegangen.

Bedingungen

122 CHANNELERR

Die RESP2-Werte lauten wie folgt:

1

Der in der Option TOCHANNEL angegebene Name enthält ein ungültiges Zeichen oder eine ungültige Zeichenkombination.

2

Der in der Option CHANNEL angegebene Kanal konnte nicht gefunden werden.

110 CONTAINERERR

Die RESP2-Werte lauten wie folgt:

1

Der in der Option TOCONTAINER angegebene Name enthält ein ungültiges Zeichen oder eine ungültige Zeichenkombination.

2

Der in der Option CONTAINER angegebene Container wurde nicht gefunden.

19 NOTOPEN

Die RESP2-Werte lauten wie folgt:

27

Das Sitzungstoken ist ungültig.

16 INVREQ

Die RESP2-Werte lauten wie folgt:

10

Der Answerheader ist ungültig.

11

Der Aktionscode ist ungültig.

13

Der Status für den Schließvorgang ist ungültig.

15

Bei der Codepagekonvertierung trat ein Fehler auf.

17

Eine Anforderung mit 'Expect-100' wurde durch den Server zurückgewiesen.

22

Die Datenblockgröße ist ungültig.

32

Der Medientyp ist ungültig.

33

Die Methode unterstützt keinen Hauptteil.

34

Die Methode erfordert einen Hauptteil.

41

Die Verbindung wurde geschlossen.

43

Der angegebene Wert für DOCSTATUS ist ungültig.

45

Der angegebene Zeichensatz ist ungültig.

- 46** Die Option CLIENTCONV ist ungültig.
- 49** Das Format der Pfadoption ist ungültig.
- 54** Die HTTP-Methode ist ungültig.
- 63** Das URIMAP-Objekt wurde inaktiviert.
- 64** Der Host in der URIMAP-Definition stimmt nicht mit dem Host überein, der beim Öffnen dieser Sitzung angegeben wurde.
- 67** Der Inhalt der Antwort entspricht nicht dem HTTP-Format. Der Fehler wird generiert, weil ein Syntaxproblem vorliegt.
- 74** Die Verbindung wurde geschlossen (CICS hat einen Header 'Connection: close' an den Server gesendet oder der Server hat möglicherweise aufgrund von Inaktivität über diese Verbindung ein zulässiges Zeitlimit überschritten).
- 76** Die Option MEDIATYPE ist erforderlich.
- 79** Das Pipelining wird gegenwärtig durchgeführt. Der Befehl WEB CONVERSE kann nicht verwendet werden.
- 80** Die Option CHARACTERSET kann nicht zusammen mit NOSRVCONVERT angegeben werden.
- 142** Der Wert für AUTHENTICATE ist ungültig. Der CVDA-Wert lautet NONE oder BASICAUTH.
- 144** Einer oder mehrere Parameter des Webbefehls sind ungültig.
- 145** Entweder wurde die Option CHANNEL nicht zusammen mit der Option CONTAINER oder die Option TOCHANNEL nicht zusammen mit der Option TOCONTAINER angegeben (und es gibt keinen aktuellen Kanal).
- 146** Der benannte Container ist schreibgeschützt.
- 147** Es trat ein interner Konvertierungsfehler auf.
- 150** Die Konvertierung wurde angefordert, aber die zu sendenden Daten befinden sich in einem Container mit dem Datentyp BIT.
- 157** Die HTTP-Header in der Antwort sind länger als 4k; daher kann CICS diese HTTP-Antwort nicht verarbeiten. Die Sitzung ist nicht verwendbar und ein Befehl **WEB CLOSE** ist erforderlich.
- 22 LENGERR**
Die RESP2-Werte lauten wie folgt:
- 5** Der Wert für die Option PATHLENGTH war nicht größer als Null.
- 8** Der Wert für die Option QUERYSTRLEN war nicht größer als Null.
- 16** Der Wert für MAXLENGTH ist ungültig.

36

Es wurde ein partieller Antworthauptteil zurückgegeben. Verwenden Sie weitere Befehle RECEIVE, um den Rest abzurufen.

50

Der Wert für die Option FROMLENGTH war nicht größer als Null.

57

Der Antworthauptteil überschreitet die angegebene Länge und der Rest des Hauptteils wurde gelöscht.

58

Der Statustext überschreitet die angegebene Länge.

59

Der Wert für die Option STATUSLEN war nicht größer als Null.

139

Der Wert für USERNAMELEN ist nicht positiv.

140

Der Wert für PASSWORDLEN ist nicht positiv.

13 NOTFND

Die RESP2-Werte lauten wie folgt:

61

Das angegebene URIMAP-Objekt wurde nicht gefunden.

112 TOKENERR

Die RESP2-Werte lauten wie folgt:

47

Das angegebene Dokumenttoken ist ungültig oder das Dokument wurde gelöscht.

17 IOERR

Die RESP2-Werte lauten wie folgt:

42

Es trat ein Socketfehler auf.

124 TIMEOUT

Die RESP2-Werte lauten wie folgt:

62

Beim Empfangen des Sockets trat eine Zeitlimitüberschreitung auf.

156

Beim Senden des Sockets trat eine Zeitlimitüberschreitung auf.

70 NOTAUTH

Die RESP2-Werte lauten wie folgt:

100

Der Pfad wurde durch den Sicherheitsexit gesperrt.

110

Es trat ein Fehler für XWBAUTH auf. Der globale Benutzerexit XWBAUTH hat einen Rückgabecode UERCERR ausgegeben, weil der Exit XWBAUTH erforderlich ist, jedoch keine gültige Antwort zurückgeben kann.

Dieser Fehlercode wird ausgegeben, wenn Folgendes zutrifft: BASICAUTH ist angegeben; USERNAME und/oder PASSWORD ist nicht angegeben; XWBAUTH ist inaktiv oder gibt die Antwort UERCERR zurück.

WEB ENDBROWSE FORMFIELD

Signalisiert das Ende einer Formularfeldsuche in einem HTML-Formular.

WEB ENDBROWSE FORMFIELD

➤ WEB — ENDBROWSE — FORMFIELD ➤

Bedingung: INVREQ

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl WEB ENDBROWSE FORMFIELD beendet den Suchvorgang nach einer Gruppe von Name/Wert-Paaren in einem HTML-Formular. Das Formular gehört zum Hauptteil einer HTTP-Anforderung, die durch die aktuelle CICS-Task verarbeitet wird. Für den Befehl ENDBROWSE werden keine Informationen zurückgegeben.

Bedingungen

16 INVREQ

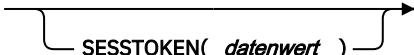
Die RESP2-Werte lauten wie folgt:

- 1 Der Befehl wurde in einer Anwendung ausgegeben, die nicht zur CICS-Webunterstützung gehört.
- 3 Der Befehl wurde für eine Anforderung ausgegeben, die keine HTTP-Anforderung ist.
- 4 Der Befehl wurde vor Ausgabe eines Befehls WEB STARTBROWSE ausgegeben.

WEB ENDBROWSE HTTPHEADER

Signalisiert das Ende eines Suchvorgangs in einem HTTP-Header.

WEB ENDBROWSE HTTPHEADER

➤ WEB — ENDBROWSE — HTTPHEADER — 

Bedingungen: INVREQ, NOTOPEN

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl WEB ENDBROWSE HTTPHEADER beendet den Suchvorgang. Für den Befehl ENDBROWSE werden keine Informationen zurückgegeben. Die Option SESSTOKEN ist erforderlich, falls die HTTP-Headerinformationen Teil einer Antwort sind, die an CICS als HTTP-Client gesendet wird.

Optionen

SESSTOKEN(*datenwert*)

Wenn CICS als HTTP-Client eingesetzt wird, ist diese Option erforderlich. Sie gibt den 8 Byte umfassenden Binärwert für das Sitzungstoken an, das eine Verbindung zwischen CICS und einem Server eindeutig kennzeichnet. Dieser Wert wird durch einen Befehl WEB OPEN für CICS als HTTP-Client zurückgegeben. Die Verwendung des Sitzungstokens ist im Abschnitt Sitzungstokens erläutert.

Bedingungen

16 INVREQ

Die RESP2-Werte lauten wie folgt:

- 1** Der Befehl wurde in einer Anwendung ausgegeben, die nicht zur CICS-Webunterstützung gehört.
 - 3** Der Befehl wurde für eine Anforderung ausgegeben, die keine HTTP-Anforderung ist.
 - 4** Der Befehl wurde vor Ausgabe eines Befehls WEB STARTBROWSE ausgegeben.
- 19 NOTOPEN**
Der RESP2-Wert lautet wie folgt:
- 27** Das Sitzungstoken ist ungültig.

WEB ENDBROWSE QUERYPARM

Beendet die Suche von Abfragezeichenfolgedaten in einer URL.

WEB ENDBROWSE QUERYPARM

➤ WEB — ENDBROWSE — QUERYPARM ➤

Bedingung: INVREQ

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl WEB ENDBROWSE QUERYPARM beendet die Suche nach einer Gruppe von Schlüsselwortparametern, die aus Name/Wert-Paaren bestehen, in einer Abfragezeichenfolge einer URL. Für den Befehl ENDBROWSE werden keine Informationen zurückgegeben.

Bedingungen

16 INVREQ

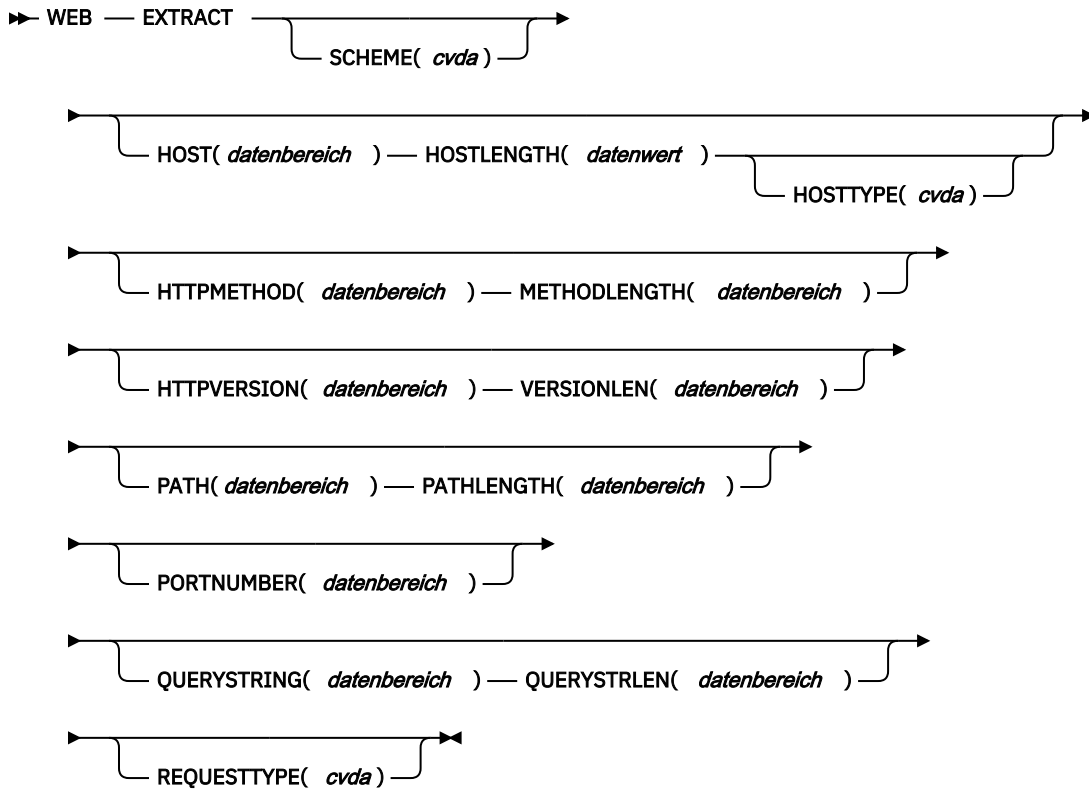
Die RESP2-Werte lauten wie folgt:

- 1** Der Befehl wurde in einer Anwendung ausgegeben, die nicht zur CICS-Webunterstützung gehört.
- 3** Der Befehl wurde für eine Anforderung ausgegeben, die keine HTTP-Anforderung ist.
- 4** Der Befehl wurde vor Ausgabe eines Befehls WEB STARTBROWSE ausgegeben.

WEB EXTRACT

Ruft Informationen zu einer HTTP-Anforderung, die an CICS als HTTP-Server ausgegeben wurde, oder zu einer Verbindung zwischen einem Internet-Server und CICS als HTTP-Client ab. Dieser Befehl ist ein Synonym für EXTRACT WEB.

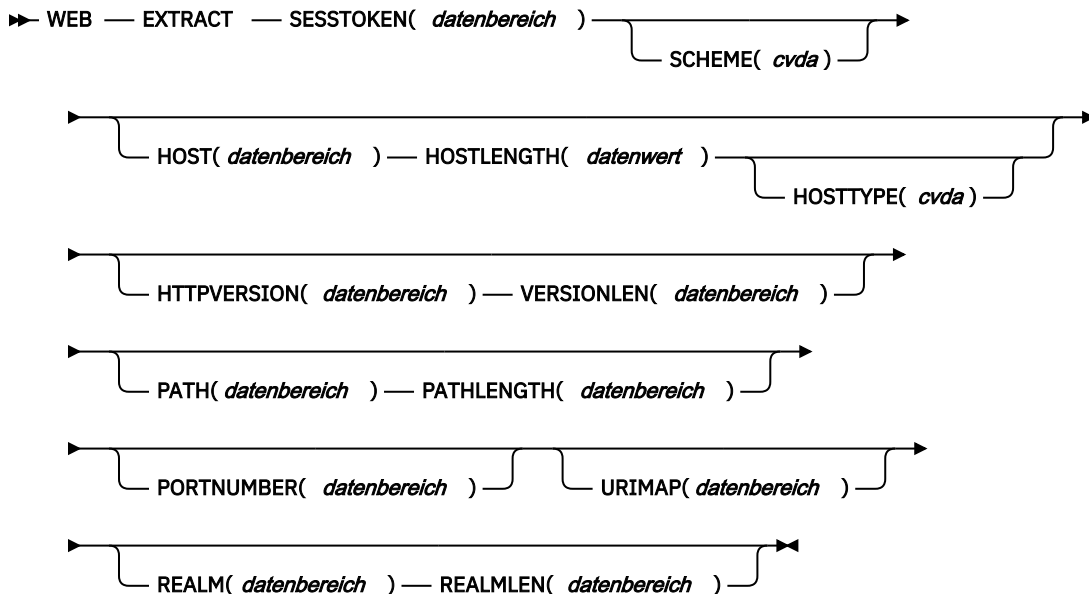
WEB EXTRACT (CICS als HTTP-Server)



Bedingungen: INVREQ, IOERR, LENGERR, NOTFND, NOTOPEN, TIMEDOUT

Dieser Befehl ist threadsicher.

WEB EXTRACT (CICS als HTTP-Client)



Bedingungen: INVREQ, IOERR, LENGERR, NOTFND, NOTOPEN, TIMEDOUT

Dieser Befehl ist threadsicher.

Beschreibung

Wenn CICS als HTTP-Server eingesetzt wird, kann eine Anwendung mit dem Befehl WEB EXTRACT Informationen zur letzten HTTP-Anforderung abrufen, die durch einen Web-Client an CICS ausgegeben und der Anwendung zur Verarbeitung zugeordnet wurde.

Wenn CICS als HTTP-Client eingesetzt wird und die Option SESSTOKEN angegeben ist, kann eine Anwendung mit dem Befehl Informationen zu einer Verbindung abrufen, die sie mit einem Server geöffnet hat. Die Informationen, die an die Anwendung zurückgegeben werden, beinhalten globale Angaben über die Verbindung wie beispielsweise den Hostnamen und die HTTP-Version des Servers. Informationen zu bestimmten Anforderungen, die durch die Anwendung ausgegeben wurden, und zu Antworten des Servers werden bei Verwendung dieses Befehls nicht zur Verfügung gestellt. Zum Abrufen von Informationen aus einer Serverantwort wird der Befehl WEB RECEIVE verwendet.

Optionen

HOST(datenbereich)

Wenn CICS als HTTP-Server eingesetzt wird, gibt die Option HOST einen Puffer an, der die Hostkomponente der URL aufnehmen soll, die entweder im Headerfeld 'Host' für die Anforderung oder in der Anforderungszeile angegeben ist (falls für die Anforderung eine absolute URI verwendet wurde). Die Portnummer wird separat mit der Option PORTNUMBER dargestellt.

Wenn CICS als HTTP-Client eingesetzt und die Option SESSTOKEN verwendet wird, gibt die Option HOST einen Puffer an, der den Hostnamen des Servers in der Verbindung aufnehmen soll, die durch die Option SESSTOKEN gekennzeichnet ist. Die Portnummer wird separat mit der Option PORTNUMBER dargestellt.

Der Hostname kann durch eine IPv4- oder eine IPv6-Adresse dargestellt werden. IPv4-Adressen werden als native IPv4-Adressen in der Schreibweise mit Trennzeichen zurückgegeben. Beispiel: 1.2.3.4. IPv6-Adressen werden als native IPv6-Hexadezimaladressen mit Doppelpunkten zurückgegeben. Beispiel: ::a:b:c:d.

Informationen zu IP-Adressen finden Sie unter [IP-Adressen](#).

HOSTLENGTH(datenbereich)

Gibt als Vollwort-Binärzahlvariable die Länge des in der Option HOST angegebenen Puffers an und wird auf die Länge der Daten gesetzt, die an die Anwendung zurückgegeben werden. Als Angabe für diesen Datenbereich ist eine Größe von 116 Zeichen zweckmäßig. Falls die Daten die Puffergröße überschreiten, tritt eine Bedingung LENGERR auf und die Daten werden abgeschnitten.

HOSTTYPE(cvda)

Gibt das Adressformat für den Wert der Option HOST zurück. Die CVDA-Werte lauten wie folgt:

HOSTNAME

Die Option HOST enthält einen aus Zeichen bestehenden Hostnamen. Die IP-Adresse, die dem Hostnamen entspricht, wird im Domänennamensserver ermittelt.

IPV4

Die Adresse ist eine IPv4-Adresse in der Schreibweise mit Trennzeichen.

IPV6

Die Adresse ist eine IPv6-Hexadezimaladresse mit Doppelpunkten.

NOTAPPLIC

Es wurde eine falsche Hostadresse zurückgegeben (HOST=0.0.0.0).

HTTPMETHOD(datenbereich)

Wenn CICS als HTTP-Server eingesetzt wird, gibt diese Option einen Puffer an, der die HTTP-Methodezeichenfolge in der Anforderungszeile der Nachricht aufnehmen soll.

Beim Einsatz von CICS als HTTP-Client ist diese Option nicht relevant.

HTTPVERSION(datenbereich)

Wenn CICS als HTTP-Server eingesetzt wird, gibt diese Option einen Puffer an, der die in der Anforderung angegebene HTTP-Version für den Web-Client aufnehmen soll.

Wenn CICS als HTTP-Client eingesetzt wird und die Option SESSTOKEN angegeben ist, gibt diese Option einen Puffer an, der die HTTP-Version des Servers in der Verbindung aufnehmen soll, die durch die Option SESSTOKEN gekennzeichnet ist. Falls CICS die HTTP-Version des Servers nicht kennt, gibt CICS an den Server eine Anforderung mit der Methode OPTIONS aus, um diese Informationen zu ermitteln.

1.1 steht für HTTP/1.1; 1.0 steht für HTTP/1.0 und früher.

METHODLENGTH(datenbereich)

Gibt als Vollwort-Binärzahlvariable die Länge des in der Option HTTPMETHOD angegebenen Puffers an und wird auf die tatsächliche Länge der Daten gesetzt, die an die Anwendung zurückgegeben werden. Falls die Daten die Puffergröße überschreiten, tritt eine Bedingung LENGERR auf und die Daten werden abgeschnitten.

PATH(datenbereich)

Wenn CICS als HTTP-Server eingesetzt wird, gibt diese Option einen Puffer an, der den Pfad in der Anforderungszeile der Nachricht aufnehmen soll.

Wenn CICS als HTTP-Client eingesetzt wird und die Option SESSTOKEN angegeben ist, gibt diese Option einen Puffer an, der den gültigen Standardpfad für Anforderungen, die mithilfe der Verbindung ausgegeben werden, aufnehmen soll. Falls eine URIMAP-Definition im Befehl WEB OPEN für die Verbindung angegeben wurde, ist der Standardpfad derjenige Pfad, der in der URIMAP-Definition festgelegt ist. Andernfalls besteht der Standardpfad aus einem einzelnen normalen Schrägstrich.

PATHLENGTH(datenbereich)

Gibt als Vollwort-Binärzahlvariable die Länge des in der Option PATH angegebenen Puffers an und wird auf die Länge der Daten gesetzt, die an die Anwendung zurückgegeben werden. Als Angabe für diesen Datenbereich ist eine Größe von 256 Zeichen zweckmäßig. Falls die Daten die Puffergröße überschreiten, tritt eine Bedingung LENGERR auf und die Daten werden abgeschnitten.

PORTNUMBER(datenbereich)

Wenn CICS als HTTP-Server eingesetzt wird, gibt diese Option einen Datenbereich zurück, der die Portnummer enthält, die in der Anforderungszeile der Nachricht angegeben ist.

Wenn CICS als HTTP-Client eingesetzt wird und die Option SESSTOKEN angegeben ist, gibt diese Option einen Datenbereich zurück, der die Portnummer enthält, mit der auf den Server in der Verbindung zugegriffen wird, die durch die Option SESSTOKEN gekennzeichnet ist.

Der im Datenbereich zurückgegebene Wert ist ein Vollwort-Binärwert.

Standardisierte Portnummern für einen Service werden in der URL normalerweise nicht angegeben. Falls die Portnummer in der URL nicht vorhanden ist, wird die Nummer vom Befehl anhand des Schemas ermittelt und zurückgegeben. Die standardisierte Portnummer für HTTP ist 80; bei HTTPS lautet die standardisierte Portnummer 443. Falls eine Portnummer zurückgegeben wird, die nicht den Standardwert für das Schema darstellt, müssen Sie die Portnummer explizit angeben, um Zugriff auf die URL zu erhalten; dies gilt beispielsweise, wenn Sie diese Informationen in einem Befehl WEB OPEN verwenden.

QUERYSTRING(datenbereich)

Wenn CICS als HTTP-Server eingesetzt wird, gibt diese Option einen Puffer an, der die Abfragezeichenfolge in der Anforderungszeile der Nachricht aufnehmen soll. Die Abfragezeichenfolge besteht aus einem oder mehreren Werten, die nach dem Fragezeichen (?) codiert sind, mit dem das Ende des Pfades begrenzt wird. Die Abfragezeichenfolge wird in ihrem Format mit Escapezeichen zurückgegeben.

Beim Einsatz von CICS als HTTP-Client ist diese Option nicht relevant.

QUERYSTRLEN(datenbereich)

Gibt als Vollwort-Binärzahlvariable die Länge des in der Option QUERY angegebenen Puffers an und wird auf die Länge der Daten gesetzt, die an die Anwendung zurückgegeben werden (die Abfragezeichenfolge). Als Angabe für diesen Datenbereich ist eine Größe von 256 Zeichen zweckmäßig. Falls die Daten die Puffergröße überschreiten, tritt eine Bedingung LENGERR auf und die Daten werden abgeschnitten.

REALM(datenbereich)

Wenn CICS als HTTP-Client eingesetzt wird, gibt diese Option den Realm oder die Sicherheitsumgebung an, in dem/der die von Ihnen angeforderten Daten enthalten sind. Falls Sie einen Befehl als Antwort auf eine HTTP-Nachricht 401 ausgeben, gibt die Option REALM den Realmwert im zuletzt empfangenen Header 'WWW-Authenticate' an.

REALMLEN(datenbereich)

Wenn CICS als HTTP-Client eingesetzt wird, gibt diese Option als Vollwort-Binärzahlvariable die Länge des Puffers an, der für die Option REALM angegeben wurde. Falls Sie einen Befehl als Antwort auf eine HTTP-Nachricht 401 ausgeben, gibt die Option REALMLEN die Länge des Realmnamens im zuletzt empfangenen Header 'WWW-Authenticate' an.

REQUESTTYPE(cvda)

Wenn CICS als HTTP-Server eingesetzt wird, gibt diese Option den Typ der empfangenen Anforderung an. Beim Einsatz von CICS als HTTP-Client ist diese Option nicht relevant. Die CVDA-Werte lauten wie folgt:

HTTPYES

Gibt an, dass es sich um eine HTTP-Anforderung handelt.

HTTPNO

Gibt an, dass es sich bei der Anforderung nicht um eine HTTP-Anforderung handelt.

SCHEME(cvda)

Diese Option gibt sowohl bei Verwendung von CICS als HTTP-Server als auch beim Einsatz von CICS als HTTP-Client (mit angegebener Option SESSTOKEN) das Schema zurück, das für die Verbindung zwischen CICS und dem Web-Client oder Web-Server verwendet wird. Die CVDA-Werte lauten wie folgt:

HTTP

Gibt an, dass das HTTP-Protokoll ohne SSL verwendet wird.

HTTPS

Gibt an, dass das HTTPS-Protokoll, also HTTP mit SSL verwendet wird.

SESSTOKEN(datenwert)

Wenn CICS als HTTP-Client eingesetzt wird, ist diese Option erforderlich. Sie gibt den 8 Byte umfassenden Binärwert für das Sitzungstoken an, das eine Verbindung zwischen CICS und einem Server eindeutig kennzeichnet. Dieser Wert wird durch einen Befehl WEB OPEN für CICS als HTTP-Client zurückgegeben. Die Verwendung des Sitzungstokens ist im Abschnitt Sitzungstokens erläutert. Für den Befehl werden Informationen zur angegebenen Verbindung zurückgegeben.

Beim Einsatz von CICS als HTTP-Server ist diese Option nicht relevant.

URIMAP(datenbereich)

Wenn CICS als HTTP-Client eingesetzt wird und die Option SESSTOKEN angegeben ist, gibt diese Option den 8 Zeichen umfassenden Namen (in Groß-/Kleinschreibung) einer URIMAP-Definition zurück, die gegebenenfalls im Befehl WEB OPEN zum Öffnen der Verbindung angegeben wurde, die durch die Option SESSTOKEN gekennzeichnet ist. Mit dem Befehl INQUIRE URIMAP können Informationen zu den Attributen dieser URIMAP-Definition ermittelt werden.

Beim Einsatz von CICS als HTTP-Server ist diese Option nicht relevant.

VERSIONLEN(datenbereich)

Gibt als Vollwort-Binärzahlvariable die Länge des in der Option HTTPVERSION angegebenen Puffers an und wird auf die Länge der Daten gesetzt, die an die Anwendung zurückgegeben werden.

Bedingungen**16 INVREQ**

RESP2-Werte:

1

Der Befehl wurde in einer Anwendung ausgegeben, die nicht zur CICS-Webunterstützung gehört.

3

Der Befehl wurde für eine Anforderung ausgegeben, die keine HTTP-Anforderung ist. (Dieser Befehl wird nur festgelegt, wenn eine oder mehrere der Optionen HTTPMETHOD, HTTPVERSION oder PATH angegeben sind und es sich bei der Anforderung nicht um eine HTTP-Anforderung handelt).

41

Die Verbindung wurde geschlossen. Dies ist ein Fehler, der nur bei Verwendung des Befehls WEB EXTRACT für einen Client auftritt. Der Server hat möglicherweise aufgrund von Inaktivität über diese Verbindung ein zulässiges Zeitlimit überschritten.

67

Der Inhalt der Antwort entspricht nicht dem HTTP-Format. Der Fehler wird generiert, weil ein Syntaxproblem vorliegt. Dieser Fehler tritt nur bei Verwendung des Befehls WEB EXTRACT für einen Client auf.

71

Beim Chunked Transfer Coding ist ein Fehler aufgetreten. Dieser Fehler tritt nur bei Verwendung des Befehls WEB EXTRACT für einen Client auf.

144

Einer oder mehrere Parameter des Webbefehls sind ungültig. Dieser Fehler tritt nur bei Verwendung des Befehls WEB EXTRACT für einen Client auf.

17 IOERR

RESP2-Werte:

42

Es trat ein Socketfehler auf.

22 LENGERR

RESP2-Werte:

4

Die Methode überschreitet die (in der Option METHODLENGTH) angegebene Länge.

5

Der Wert für die Option PATHLENGTH war nicht größer als Null.

6

Die HTTP-Version überschreitet die (in der Option VERSIONLEN) angegebene Länge.

7

Der Wert für die Option VERSIONLEN war nicht größer als Null.

8

Die Abfragezeichenfolge überschreitet die (in der Option QUERYSTRLEN) angegebene Länge.

21

Der Wert für die Option HOSTLENGTH war nicht größer als Null.

29

Der Hostname überschreitet die (in der Option HOSTLENGTH) angegebene Länge.

30

Der Pfad überschreitet die (in der Option PATHLENGTH) angegebene Länge.

141

Der Wert für REALMLen ist nicht positiv oder nicht groß genug für den in der HTTP-Antwort 401 zurückgegebenen Realmwert.

13 NOTFND

RESP2-Werte:

155

Die Informationen zur Anforderungszeile wurden nicht gefunden.

19 NOTOPEN

RESP2-Werte:

27

Das Sitzungstoken ist ungültig.

124 TIMEDOUT

RESP2-Werte:

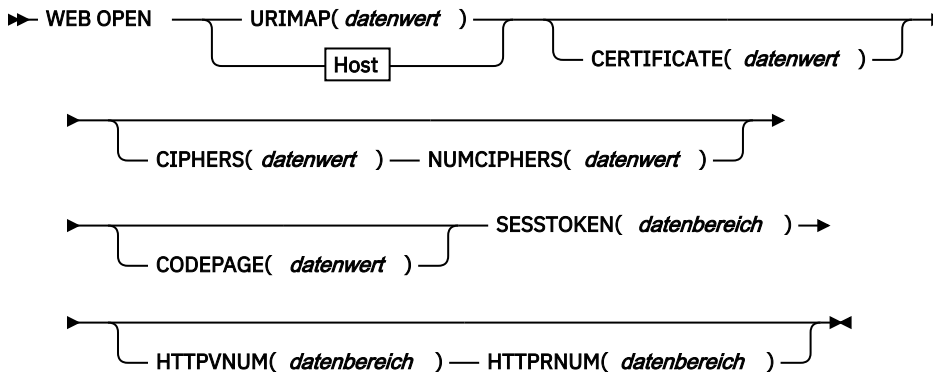
62

Beim Empfangen des Sockets trat eine Zeitlimitüberschreitung auf.

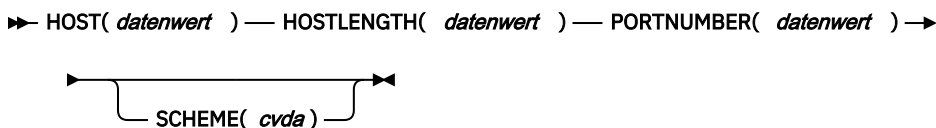
WEB OPEN

Öffnet eine Verbindung zu einem Server für CICS als HTTP-Client.

WEB OPEN



Host



Bedingungen: IOERR, INVREQ, LENGERR, NOTFND, NOTAUTH, TIMEDOUT

Dieser Befehl ist threadsicher.

Beschreibung

Mit dem Befehl **WEB OPEN** kann ein Anwendungsprogramm über die CICS-Webunterstützung eine Verbindung zu einem angegebenen Host auf einem HTTP-Server im Internet herstellen. Sobald die Verbindung geöffnet wurde, kann das Anwendungsprogramm HTTP-Clientanforderungen an den Server ausgeben und Antworten von ihm empfangen.

Beim Öffnen der Verbindung können Sie eine URIMAP-Ressource angeben, die die Informationen zur URL für die Verbindung enthält. Diese Informationen können Sie direkt im Befehl **WEB OPEN** angeben, statt eine URIMAP-Ressource zu verwenden. Die Verwendung einer URIMAP-Ressource bietet allerdings die folgenden Vorteile:

- Etwaige Änderungen am Endpunkt der Verbindung können von Systemadministratoren verwaltet werden, sodass Sie Ihre Anwendungen nicht erneut kompilieren müssen, falls sich die URL für Ihre Anforderung ändert.
- Bei Verwendung von SSL können Sie in der URIMAP-Ressource ein SSL-Clientzertifikat oder Cipher-Suite-Codes angeben, damit Systemadministrator Änderungen an diesen Zertifikaten und Codes verwalten können.
- Sie können auswählen, dass CICS die Verbindungen, die mit der URIMAP-Ressource geöffnet wurden, nach ihrer Verwendung geöffnet lässt und sie zur Wiederverwendung durch eine andere Anwendung bzw. eine andere Instanz derselben Anwendung in einen Pool stellt. Das Verbindungspooling ist nur verfügbar, wenn Sie eine URIMAP-Ressource angeben deren Attribut SOCKETCLOSE festgelegt ist. Weitere

Informationen zu den Leistungsvorteilen, die sich durch das Verbindungspooling ergeben, finden Sie unter [Verbindungspooling für HTTP-Clientleistung](#).

Angaben über die Erstellung einer URIMAP-Ressource für eine Clientanforderung enthält der Abschnitt [URIMAP-Ressource für CICS als HTTP-Client erstellen](#).

Der Befehl **WEB OPEN** steuert den Benutzerexit XWBOPEN, der die Verbindung zum Server bei Bedarf über einen Proxy-Server erfolgen lassen kann.

Anmerkung: Falls die Verbindungsanforderung innerhalb des Deadlock-Zeitlimitintervalls (angegeben im Attribut DTIMOUT der Transaktionsdefinition für die Transaktion, die die Benutzeranwendung startet) nicht aufgebaut werden kann, gibt CICS eine Antwort TIMEDOUT an die Anwendung zurück. Die Festlegung der Einstellung NO für DTIMOUT oder die zugelassene Verwendung von NO als Standardwert bedeutet, dass die Anwendung unendlich warten kann.

Optionen

CERTIFICATE(datenwert)

Gibt die Bezeichnung des X.509-Zertifikats an, das beim SSL-Handshake als SSL-Clientzertifikat verwendet werden soll. Zertifikatsbezeichnungen können bis zu 32 alphanumerische Zeichen umfassen. Diese Option ist nur bei Angabe der Option HTTPS relevant. Falls die Option HTTPS, jedoch nicht die Option CERTIFICATE angegeben ist, wird das Standardzertifikat verwendet, das im Schlüsselring für die Benutzer-ID der CICS-Region definiert ist. Das Zertifikat muss in einem Schlüsselring in der Datenbank des externen Sicherheitsmanagers gespeichert sein. Weitere Informationen finden Sie unter [Schlüsselring manuell erstellen](#).

CIPHERS(datenwert)

Gibt eine aus bis zu 56 Hexadezimalziffern bestehende Zeichenfolge an, die als Liste von bis zu 28 zweistelligen Cipher-Suite-Codes interpretiert wird. Die Cipher-Suite-Codes werden verwendet, wenn SSL für die Verbindung aktiv ist; diese Option ist daher nur bei Angabe der Option HTTPS relevant. Die Codes geben die Verschlüsselungsmethode an, die für diese Verbindung zu verwenden ist.

Verwenden Sie die Option NUMCIPHERS, um die Anzahl der Cipher-Suite-Codes in Ihrer Liste anzugeben. Welche Codes verfügbar sind, ist von der Verschlüsselungsstufe abhängig, die durch den Initialisierungsparameter ENCRYPTION angegeben wurde. Falls Sie Verschlüsselungscodes angeben, die nicht in der Standardliste für die aktive Verschlüsselungsstufe enthalten sind, werden diese Codes ignoriert. Weitere Informationen zu Cipher-Suites enthält der Abschnitt [Cipher-Suites und Cipher-Suite-Spezifikationsdateien](#).

Sie können die Option URIMAP angeben, um diese Informationen direkt aus einer vorhandenen URIMAP-Definition zu verwenden; in diesem Fall ist die Option CIPHERS nicht erforderlich. Trotzdem können Sie die Option CIPHERS angeben; dann wird die Einstellung in der URIMAP-Definition durch alle Codes überschrieben, die Sie für diese Option angeben.

Anmerkung: Der EXEC CICS-Befehl **WEB OPEN** unterstützt nicht die Verwendung einer Spezifikationsdatei für eine SSL-Cipher-Suite in der Option CIPHERS.

Falls Sie die Optionen CIPHERS und URIMAP nicht angeben, SSL jedoch für die Verbindung aktiv ist, wird die Standardschlüsselliste für die Verschlüsselungsstufe des aktiven Systems verwendet.

CODEPAGE(datenwert)

Gibt eine Codepage an, die für das Anwendungsprogramm geeignet ist. Der Name der Codepage kann bis zu 8 alphanumerische Zeichen umfassen. Standardwert ist die Standardcodepage für die lokale CICS-Region, die im Systeminitialisierungsparameter LOCALCCSID angegeben war. Die Codepage gilt für die Dauer dieser Verbindung. Wenn der Server eine Antwort auf eine HTTP-Anforderung zurückgibt und die Konvertierung angefordert wurde (dies ist die Standardeinstellung), konvertiert CICS den Anforderungshauptteil in diese Codepage, bevor er ihn an die Anwendung übergibt.

Das CICS-Standardformat für diesen Codepagenamen besteht aus der Codepagenummer (oder allgemeiner der CCSID), die nach Bedarf mithilfe von 3 bis 5 Dezimalziffern geschrieben und anschließend mit nachgestellten Leerzeichen auf 8 Zeichen aufgefüllt wird. Bei der Codepage 37, die aus weniger als 3 Ziffern besteht, lautet das Standardformat '037'. CICS akzeptiert als Codepagenamen auch

eine beliebige Dezimalzahl mit bis zu 8 Ziffern (aufgefüllt mit nachgestellten Leerzeichen) im Bereich von 1 bis 65535, auch wenn dieser nicht im Standardformat angegeben ist.

HOST(datenwert)

Gibt den Hostnamen des Servers an, zu dem Sie eine Verbindung herstellen wollen. Diese Informationen können Sie mit dem Befehl **WEB PARSE URL** aus einer bekannten URL oder mit dem Befehl **WEB EXTRACT URIMAP** aus einer vorhandenen URIMAP-Definition extrahieren. Durch Angabe der Option URIMAP können Sie diese Angaben direkt aus einer vorhandenen URIMAP-Definition verwenden; in diesem Fall ist die Option HOST nicht erforderlich. HTTP-Clientverbindungen können nur dann zur Wiederverwendung in den Pool gestellt werden, wenn Sie die Option URIMAP angeben. Die Verwendung der Option HOST ermöglicht selbst dann kein Verbindungspooling, wenn Sie die Informationen aus einer URIMAP-Definition extrahieren.

Wenn Transport Layer Security (TLS) für die Verbindung zu einem HTTP-Server verwendet wird, übergibt CICS den Hostnamen beim TLS-Handshake in in der SNI-Erweiterung (SNI = Server Name Indication). Dadurch kann CICS die TLS-Kommunikation mit einem virtuellen Host verwenden, wenn der Server mehrere virtuelle Hosts unterstützt, die eine einzelne IP-Adresse nutzen. Weitere Informationen finden Sie unter [Grundlagen der CICS-Webunterstützung: Virtuelles Hosting](#).

Der Hostname kann durch einen aus Zeichen bestehenden Hostnamen, eine IPv4-Adresse oder eine IPv6-Adresse dargestellt werden. Stellen Sie bei Angabe einer IPv6-Adresse (oder eines Hostnamens, der in eine IPv6-Adresse aufgelöst wird) sicher, dass Sie in einer Umgebung mit Dualmodus (IPv4 und IPv6) arbeiten und dass der Client oder Server, mit dem Sie kommunizieren, ebenfalls in einer Umgebung mit Dualmodus (IPv4 und IPv6) betrieben wird.

Weitere Informationen zu IPv6 finden Sie unter [Hinweise zu IPv6 und CICS](#).

Zur Angabe von IPv4- und IPv6-Adressen können Sie eine Reihe von Formaten verwenden. Informationen zu IP-Adressen finden Sie unter [IP-Adressen](#).

Falls eine Portnummer erforderlich ist, dürfen Sie die Portnummer nicht in den Wert für die Option HOST einbeziehen. Verwenden Sie stattdessen die Option PORTNUMBER.

Zulässige Zeichen:

Alphanumerische Zeichen, Bindestriche (-), Doppelpunkte (:) oder Punkte (.)

Doppelpunkte (:) können ausschließlich in der IP-Adresse verwendet werden. Die Verwendung von Doppelpunkten (:) in einem aus Zeichen bestehenden Hostnamen ist nicht möglich.

HOSTLENGTH(datenwert)

Gibt als Vollwort-Binärwert die Länge des Hostnamens an. Diese Informationen werden zurückgegeben, wenn Sie mit dem Befehl **WEB PARSE URL** eine URL analysieren. Sie können außerdem mit dem Befehl **WEB EXTRACT URIMAP** aus einer vorhandenen URIMAP-Definition extrahiert werden. Durch Angabe der Option URIMAP können Sie diese Angaben direkt aus einer vorhandenen URIMAP-Definition verwenden; in diesem Fall ist die Option HOSTLENGTH nicht erforderlich.

HTTPRNUM(datenbereich)

Gibt als Halbwort-Binärwert die Releasenummer für die HTTP-Version des Servers zurück. (Die Option HTTPVNUM gibt die Versionsnummer zurück.) Hat der Server beispielsweise den Stand von HTTP/1.0, gibt die Option HTTPRNUM den Wert 0 zurück.

HTTPVNUM(datenbereich)

Gibt als Halbwort-Binärwert die Versionsnummer für die HTTP-Version des Servers zurück. (Die Option HTTPRNUM gibt die Releasenummer zurück.) Hat der Server beispielsweise den Stand von HTTP/1.0, gibt die Option HTTPVNUM den Wert 1 zurück.

Falls Sie die Optionen HTTPVNUM und HTTPRNUM angeben, ruft CICS die Informationen zur HTTP-Version ab, sobald die Verbindung zum Server geöffnet wird. Wenn der Server als Antwort auf diese Anforderung keine Informationen zur HTTP-Version bereitstellt oder die Version älter als 1.0 ist, geht CICS von der Version HTTP/1.0 aus.

Geben Sie diese Optionen an, wenn Sie sich durch eine Überprüfung der Informationen zur HTTP-Version unbedingt vergewissern müssen, dass eine geplante Aktion durch Ihre Anwendung vor oder wäh-

rend der ersten Anforderung erfolgreich verläuft. Von der HTTP-Version sind unter anderem folgende Aktionen abhängig:

- HTTP-Header schreiben, mit denen eine Aktion angefordert wird, die von einem Server mit einer älteren Version als HTTP/1.1 möglicherweise nicht korrekt ausgeführt wird
- HTTP-Methoden verwenden, die für Server mit einer älteren Version als HTTP/1.1 möglicherweise nicht geeignet sind
- Chunked Transfer Coding verwenden
- Pipelinefolge von Anforderungen senden

Die zusätzliche HTTP-Anforderung, mit der CICS die Informationen zur HTTP-Version anfordert, hat Einfluss auf das Leistungsverhalten; geben Sie diese Optionen daher nicht an, wenn es zum gegenwärtigen Zeitpunkt nicht erforderlich ist. Sobald die erste Antwort vom Server empfangen wurde, können Sie diese Angaben mit dem Befehl WEB EXTRACT abrufen.

NUMCIPHERS(datenwert)

Gibt als Halbwort-Binärwert die Anzahl der Cipher-Suite-Codes an, die Sie für die Option CIPHERS angegeben haben.

PORTNUMBER(datenwert)

Gibt als Vollwort-Binärwert die Portnummer an. Die Portnummer geben Sie nur dann an, wenn es sich *nicht* um den Standardwert für das angegebene Schema handelt. Die Standardportnummer für HTTP ist 80; bei HTTPS lautet die Standardportnummer 443. Informationen zur Portnummer können Sie mit dem Befehl WEB PARSE URL aus einer bekannten URL oder mit dem Befehl WEB EXTRACT URIMAP aus einer vorhandenen URIMAP-Definition extrahieren. Durch Angabe der Option URIMAP können Sie diese Angaben direkt aus einer vorhandenen URIMAP-Definition verwenden; in diesem Fall ist die Option PORTNUMBER nicht erforderlich. HTTP-Clientverbindungen können nur dann zur Wiederverwendung in den Pool gestellt werden, wenn Sie die Option URIMAP angeben. Die Verwendung der Option PORTNUMBER ermöglicht selbst dann kein Verbindungspooling, wenn Sie die Informationen aus einer URIMAP-Definition extrahieren.

SCHEME(cvda)

Gibt das Schema an, das für die Verbindung zum Server, die mit oder ohne SSL ausgeführt werden kann, zu verwenden ist. Die CVDA-Werte lauten wie folgt:

HTTP

Gibt an, dass das HTTP-Protokoll ohne SSL verwendet wird.

HTTPS

Gibt an, dass das HTTPS-Protokoll, also HTTP mit SSL verwendet wird. Bei Verwendung von HTTPS muss der CICS-Adressraum für SSL aktiviert sein.

Diese Informationen können Sie mit dem Befehl WEB PARSE URL aus einer bekannten URL oder mit dem Befehl WEB EXTRACT URIMAP aus einer vorhandenen URIMAP-Definition extrahieren. Durch Angabe der Option URIMAP können Sie diese Angaben direkt aus einer vorhandenen URIMAP-Definition verwenden; in diesem Fall ist die Option SCHEME nicht erforderlich. HTTP-Clientverbindungen können nur dann zur Wiederverwendung in den Pool gestellt werden, wenn Sie die Option URIMAP angeben. Die Verwendung der Option SCHEME ermöglicht selbst dann kein Verbindungspooling, wenn Sie die Informationen aus einer URIMAP-Definition extrahieren.

SESTOKEN(datenbereich)

Gibt den 8 Byte umfassenden Binärwert für das Sitzungstoken zurück, der die Nutzung der Verbindung zwischen CICS und dem Server durch diese Anwendung eindeutig kennzeichnet. Das Sitzungstoken muss bei allen CICS-Befehlen WEB verwendet werden, die sich auf diese Verbindung beziehen. Informationen zur Verwendung des Sitzungstokens finden Sie unter [Sitzungstokens](#).

URIMAP(datenwert)

Gibt den aus bis zu 8 Zeichen in Groß-/Kleinschreibung bestehenden Namen einer URIMAP-Definition an, von der die folgenden Informationen bereitgestellt werden:

- Das Schema, das für die Verbindung zum Server zu verwenden ist.

- Der Hostname des Servers, zu dem Sie eine Verbindung herstellen wollen.
- Eine Portnummer, sofern erforderlich.
- Eine Pfadkomponente für die URI, die die Ressource auf dem Server darstellt, auf die Sie zugreifen wollen. Dieser Pfad wird zum Standardpfad für Befehle WEB SEND oder WEB CONVERSE, die sich auf die Verbindung beziehen, kann jedoch durch Angabe eines anderen Pfades im Befehl WEB SEND oder WEB CONVERSE überschrieben werden.
- Der Ablaufzeitraum für gepoolte Verbindungen, die unter Verwendung der URIMAP-Ressource geöffnet wurden. Das Verbindungspooling ist aktiviert, wenn Sie in der URIMAP-Definition mit dem Attribut SOCKETCLOSE einen Ablaufzeitraum festlegen und die URIMAP-Ressource im Befehl WEB OPEN angeben.
- Die Bezeichnung des X.509-Zertifikats an, das bei Bedarf als SSL-Clientzertifikat verwendet werden soll.
- Die Cipher-Suite-Codes, die für die Verbindung verwendet werden können.

Geben Sie bei Angabe der Option URIMAP nicht die Optionen CERTIFICATE, HOST, HOSTLENGTH, PORTNUMBER, PORTLENGTH oder SCHEME an. Die Optionen CIPHERS und NUMCIPHERS können im Befehl angegeben oder weggelassen werden; sind sie angegeben, überschreiben ihre Werte die entsprechenden Einstellungen in der URIMAP-Definition.

Anmerkung: Der EXEC CICS-Befehl **WEB OPEN** unterstützt nicht die Verwendung einer Spezifikationsdatei für eine SSL-Cipher-Suite in der Option CIPHERS.

Die URIMAP-Definition muss für CICS als HTTP-Client, mit Angabe von USAGE(CLIENT), bestimmt sein.

Bedingungen

17 IOERR

RESP2-Werte:

38

Es trat ein Fehler für den Proxy auf.

42

Es trat ein Socketfehler auf. Es liegt ein Konnektivitätsproblem vor.

16 INVREQ

RESP2-Werte:

14

Die Codepage ist falsch.

22

Während der ersten HTTP-Anforderung mit Verwendung der Methode OPTIONS wurde ein falscher Block empfangen.

23

Das Clientzertifikat ist falsch.

40

Das Schema ist falsch.

41

Der Server hat die Verbindung während der ersten HTTP-Anforderung mit Verwendung der Methode OPTIONS geschlossen.

48

Das Format der Hostoption ist falsch.

63

Das URIMAP-Objekt ist nicht verfügbar.

66

Bei der Verarbeitung für den Exit XWBOPEN trat ein Fehler auf.

67

Der Inhalt der Antwort entspricht nicht dem HTTP-Format. Der Fehler wird generiert, weil ein Syntaxproblem vorliegt.

96

SSL wird nicht unterstützt.

137

Alle angeforderten Verschlüsselungscodes wurden abgelehnt.

138

Die Portnummer ist größer als 65535.

144

Einer oder mehrere Parameter des Webbefehls sind falsch.

22 LENGERR

RESP2-Werte:

21

Die Länge des Hostwertes ist falsch.

13 NOTFND

RESP2-Werte:

20

Der Hostname wird durch den Namensserver nicht aufgelöst oder das Format der Hostoption ist falsch.

39

Der Proxy ist unbekannt.

61

Das angegebene URIMAP-Objekt wurde nicht gefunden.

70 NOTAUTH

RESP2-Werte:

100

Der Hostname wurde durch den Sicherheitsexit gesperrt.

124 TIMEDOUT

RESP2-Werte:

62

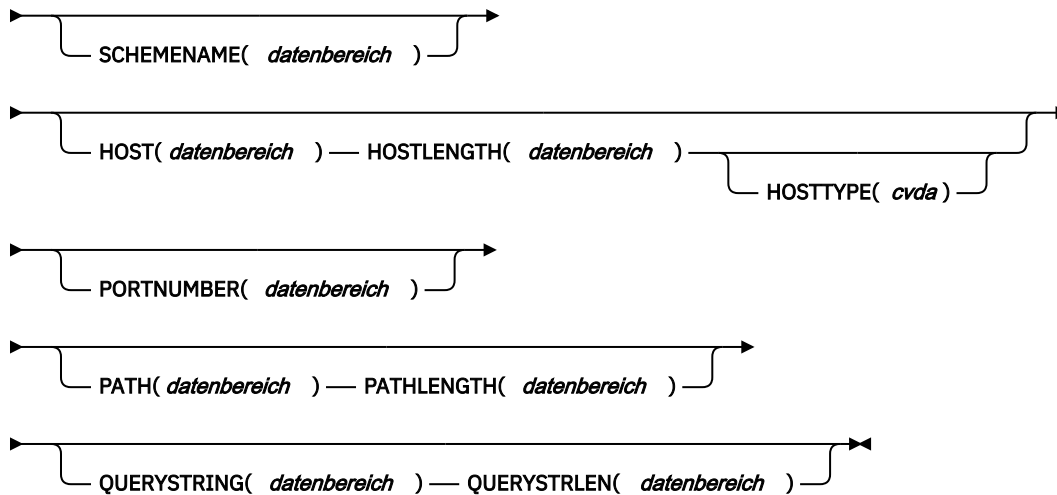
Beim Empfangen des Sockets trat eine Zeitlimitüberschreitung auf.

WEB PARSE URL

Gliedert eine URL-Zeichenfolge in ihre einzelnen Bestandteile auf.

WEB PARSE URL

➔ WEB — PARSE URL(*datenwert*) — URLENGTH(*datenwert*) →



Bedingungen: INVREQ, LENGERR

Dieser Befehl ist threadsicher.

Beschreibung

Mit dem Befehl WEB PARSE URL können Sie eine URL-Zeichenfolge in ihre einzelnen Bestandteile wie Schema, Host, Port, Pfad und Abfragezeichenfolge, die auch als 'Komponenten' bezeichnet werden, aufgliedern. Diese Bestandteile sind im Abschnitt [Komponenten einer URL](#) erläutert. Mithilfe dieses Prozesses können Sie den Aufbau der URL untersuchen und die einzelnen Bestandteile erkennen. Die zurückgegebenen Informationen können im Befehl WEB OPEN verwendet werden, um eine Clientverbindung zu dem in der URL benannten Host zu öffnen.

Alle gegebenenfalls in der URL gefundenen Escapezeichenfolgen werden auf ihre Gültigkeit überprüft. Eine Escapezeichenfolge besteht aus einem Prozentzeichen (%), auf das zwei Hexadezimalzeichen folgen. Gültige Hexadezimalzeichen sind die Ziffern 0 bis 9 und die Buchstaben A bis F.

Wenn die Zeichenfolgeneingabe für den Befehl WEB PARSE URL auf eine für URLs korrekte Weise begrenzt wurde, ist zu beachten, dass falscher Inhalt durch den Befehl nicht erkannt wird, beispielsweise ein Hostname, der keinen vorhandenen Host im Internet darstellt, oder ein Zeichen, das in einer URL nicht zulässig ist.

Optionen

HOST(*datenbereich*)

Gibt die Hostkomponente der URL zurück. Dieser Wert kann entweder ein aus Zeichen bestehender Hostname oder eine numerische IP-Adresse sein. Falls in der URL explizit eine Portnummer angegeben ist, wird die Portnummer in der Option PORTNUMBER separat zurückgegeben.

Der Hostname kann durch eine IPv4- oder eine IPv6-Adresse dargestellt werden. IPv6-Adressen werden als native IPv6-Hexadezimaladressen mit Doppelpunkten zurückgegeben. Beispiel: ::a:b:c:d. Falls Sie eine IPv6-Adresse in einer URL angeben (z. B. `http://[::a:b:c:d]:80`), gibt die Option HOST die Adresse ohne eckige Klammern zurück.

Verwenden Sie zur Darstellung von eckigen Klammern bei der Angabe von IPv6-Adressen die Zeichen X'BA' und X'BB' (Codepage 37).

Informationen zu IP-Adressen finden Sie unter [IP-Adressen](#).

HOSTLENGTH(*datenbereich*)

Gibt als Vollwort-Binärzahlvariable die Länge des in der Option HOST angegebenen Puffers an und wird auf die Länge der Daten (für den Hostnamen) gesetzt, die an die Anwendung zurückgegeben wer-

den. Als Angabe für diesen Datenbereich wird eine Größe von 116 Zeichen als zweckmäßig empfohlen. Falls die Daten die Puffergröße überschreiten, tritt eine Bedingung LENGERR auf und die Daten werden abgeschnitten.

HOSTTYPE(*cvda*)

Gibt das Adressformat für den Wert der Option HOST zurück. Die CVDA-Werte lauten wie folgt:

HOSTNAME

Die Option HOST enthält einen aus Zeichen bestehenden Hostnamen. Die IP-Adresse, die dem Hostnamen entspricht, wird im Domänennamensserver ermittelt.

IPV4

Die Adresse ist eine IPv4-Adresse in der Schreibweise mit Trennzeichen.

IPV6

Die Adresse ist eine IPv6-Hexadezimaladresse mit Doppelpunkten.

PATH(*datenbereich*)

Gibt die Pfadkomponente der URL zurück.

PATHLENGTH(*datenbereich*)

Gibt als Vollwort-Binärzahlvariable die Länge des in der Option PATH angegebenen Puffers an und wird auf die tatsächliche Länge der Daten (die Pfadkomponente der URL) gesetzt, die an die Anwendung zurückgegeben werden. Als Angabe für diesen Datenbereich wird eine Größe von 256 Zeichen als zweckmäßig empfohlen. Falls die Daten die Puffergröße überschreiten, tritt eine Bedingung LENGERR auf und die Daten werden abgeschnitten.

PORTNUMBER(*datenbereich*)

Gibt als Datenbereich von Vollwort-Binärzahlen die Portnummer zurück, die in der URL angegeben bzw. für sie geeignet ist. Portnummern werden manchmal in einer URL nach dem Hostnamen explizit angegeben. Standardisierte Portnummern für einen Service werden jedoch in der URL normalerweise nicht angegeben. Falls die Portnummer in der URL nicht vorhanden ist, wird die Nummer vom Befehl WEB PARSE URL anhand des Schemas ermittelt und zurückgegeben. Die standardisierte Portnummer für HTTP ist 80; bei HTTPS lautet die standardisierte Portnummer 443. Falls eine Portnummer zurückgegeben wird, die nicht den Standardwert für das Schema darstellt, müssen Sie die Portnummer explizit angeben, um Zugriff auf die URL zu erhalten, dies gilt beispielsweise, wenn Sie diese Informationen in einem Befehl WEB OPEN verwenden.

QUERYSTRING(*datenbereich*)

Gibt die Abfragezeichenfolge aus der URL zurück. Die Abfragezeichenfolge besteht aus einem oder mehreren Werten, die nach dem Fragezeichen (?) codiert sind, mit dem das Ende des Pfades begrenzt wird. Die Abfragezeichenfolge wird in ihrem Format mit Escapezeichen zurückgegeben.

QUERYSTRLEN(*datenbereich*)

Gibt als Vollwort-Binärzahlvariable die Länge des in der Option QUERYSTRING angegebenen Puffers an und wird auf die Länge der Daten gesetzt, die an die Anwendung zurückgegeben werden (die Abfragezeichenfolge). Als Angabe für diesen Datenbereich wird eine Größe von 256 Zeichen als zweckmäßig empfohlen. Falls die Daten die Puffergröße überschreiten, tritt eine Bedingung LENGERR auf und die Daten werden abgeschnitten.

SCHEMENAME(*datenbereich*)

Gibt die Schemakomponente der URL in Form eines 16 Zeichen umfassenden Datenbereichs zurück. Nur die Schemas HTTP und HTTPS (also das HTTP-Protokoll mit und ohne SSL) werden durch CICS unterstützt und können in einem Befehl WEB OPEN verwendet werden.

Der Schemaname wird immer in Großbuchstaben zurückgegeben.

URL(*datenwert*)

Gibt die vollständige URL-Zeichenfolge an.

URLLENGTH(*datenwert*)

Gibt als Vollwort-Binärwert die Länge des Puffers an, der die URL-Zeichenfolge enthält.

Bedingungen

16 INVREQ

RESP2-Werte:

28

Die URL ist falsch.

65

Die Escapezeichenfolge ist fehlerhaft.

22 LENGERR

RESP2-Werte:

8

Die Länge der zurückgegebenen Abfragezeichenfolge ist größer als der Wert von QUERYSTRLEN.

29

Die Länge des zurückgegebenen Hostnamens ist größer als der Wert von HOSTLENGTH.

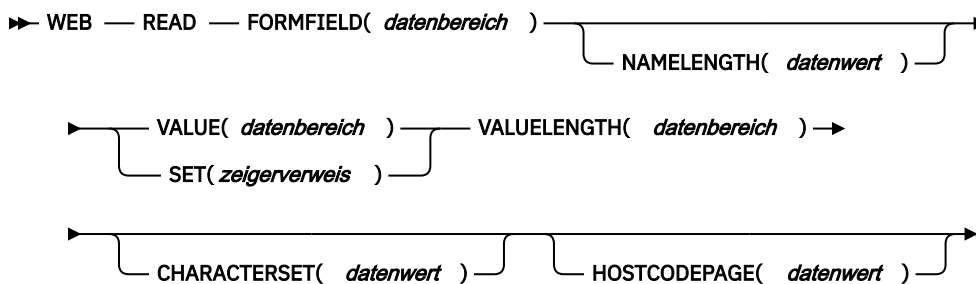
30

Die Länge des zurückgegebenen Pfades ist größer als der Wert von PATHLENGTH.

WEB READ FORMFIELD

Ruft den Wert eines Felds aus einem HTML-Formular ab.

WEB READ FORMFIELD



Bedingungen: INVREQ, LENGERR, NOTFND

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl `WEB READ FORMFIELD` ruft den Wert eines bestimmten Feldes aus einem HTML-Formular ab. Der Name des Formularfeldes ist im Parameter `FORMFIELD` angegeben. Die Formulardaten werden als Teil einer HTTP-Anforderung gesendet, die durch die aktuelle CICS-Task verarbeitet wird.

Der Web-Client sendet Formulardaten in einer Abfragezeichenfolge, wenn die Methode GET verwendet wird, bzw. im Entitätshauptteil, wenn die Methode POST verwendet wird. CICS kann die Daten aus diesen beiden Positionen extrahieren.

Die Formulardaten werden in ihrem Format ohne Escapezeichen zurückgegeben (eine Erläuterung finden Sie unter [Reservierte und ausgeschlossene Zeichen](#)).

Falls die empfangenen Daten eine Datei darstellen, wird die hochgeladene Datei *keiner* Codepagekonvertierung unterzogen.

CICS liest Formulardaten nur dann, wenn CICS die Rolle des HTTP-Servers innehat. Beim Einsatz von CICS als HTTP-Client ist die Funktion nicht verfügbar.

Optionen

CHARACTERSET(name)

Gibt den 40 Zeichen langen Namen des Zeichensatzes an, der zur Codierung der Formulardaten erforderlich ist. Diese Option sollte mit der Formularcodierung übereinstimmen, die durch das entsprechende HTML-Formular festgelegt ist (weitere Informationen finden Sie unter [Vorgehensweise zur Bestimmung der Clientcodierung](#)). CICS unterstützt nicht alle von IANA genannten Zeichensätze. Im Abschnitt [HTML-codierte Zeichensätze](#) sind die IANA-Zeichensätze aufgelistet, die durch CICS für die Codepagekonvertierung unterstützt werden.

CLNTCODEPAGE(name)

Diese Option wird nur zu Upgradezwecken unterstützt. Sie wurde durch die Option CHARACTERSET ersetzt. Bei beiden Schlüsselwörtern führt CICS dieselbe Aktion aus.

FORMFIELD(datenbereich)

Gibt den Namen des Formularfeldes an, das extrahiert werden soll. Der Wert ist eine Textzeichenfolge, die den Namen des angeforderten Feldes enthält. Die Groß-/Kleinschreibung wird bei der Textzeichenfolge nicht beachtet.

HOSTCODEPAGE(datenwert)

Gibt den 8 Zeichen langen Namen der CICS-Codepage (für den Host) an, die durch das Anwendungsprogramm erforderlich ist und in die die Formulardaten konvertiert werden sollen. Normalerweise handelt es sich hierbei um eine EBCDIC-Codepage.

Das CICS-Standardformat für einen Codepagenamen besteht aus der Codepagenummer (oder allgemeiner der CCSID), die nach Bedarf mithilfe von 3 bis 5 Dezimalziffern geschrieben und anschließend mit nachgestellten Leerzeichen auf 8 Zeichen aufgefüllt wird. Bei der Codepage 37, die aus weniger als 3 Ziffern besteht, lautet das Standardformat '037'. CICS akzeptiert jetzt als Codepagenamen auch eine beliebige Dezimalzahl mit bis zu 8 Ziffern (aufgefüllt mit nachgestellten Leerzeichen) im Bereich von 1 bis 65535, auch wenn dieser nicht im Standardformat angegeben ist.

Falls die Codepage nicht angegeben ist, werden die Daten in der EBCDIC-Codepage zurückgegeben, die im Systeminitialisierungsparameter LOCALCCSID angegeben ist (der für die lokale CICS-Region gilt und den Standardwert 037 besitzt), unter der Voraussetzung, dass die angegebene Codepage durch die CICS-Webschnittstelle unterstützt wird. Die Codepage wird unterstützt, wenn sie in einer Liste von EBCDIC-Codepages angegeben ist, die von CICS als ausreichender Standard für eine erfolgreiche Analyse von Web-Headern anerkannt werden (hierzu zählen alle SBCS-CECP und Euro-Codepages). Andernfalls gibt CICS die Daten stattdessen in der EBCDIC-Standardcodepage 037 zurück.

NAMELENGTH(datenwert)

Gibt als Vollwort-Binärwert die Länge des Formularfeldnamens an.

SET(zeigerverweis)

Gibt einen Zeigerverweis an, der auf die Adresse der empfangenen Daten gesetzt werden soll. Der Zeigerverweis ist bis zum Ende der Task gültig.

VALUE(datenbereich)

Gibt den Puffer an, der den Wert des benannten Formularfeldes aufnehmen soll. CICS entfernt alle etwaigen Escapezeichen bei Zeichen mit Escapezeichen, bevor sie in den Puffer gestellt werden.

VALUELENGTH(datenbereich)

Gibt als Vollwort-Binärwert die Länge des Formularfeldwertes an. In diesem Datenbereich wird die tatsächliche Länge des Wertes zurückgegeben. Falls Sie die Option VALUE angeben, gibt die Option VALUELENGTH die maximale Länge der Daten an, die das Programm akzeptiert. Wenn der Wert die Länge des Puffers überschreitet, wird er abgeschnitten. Ist die Länge des Formularfeldwertes kleiner als die Größe des Puffers, wird der Formularfeldwert an die äußeren linken Bytepositionen gestellt.

Bedingungen

16 INVREQ

Die RESP2-Werte lauten wie folgt:

1

Der Befehl wurde in einer Anwendung ausgegeben, die nicht zur CICS-Webunterstützung gehört.

Wenn Sie CICS als HTTP-Server nutzen, können Sie den Befehl WEB READ HTTPHEADER entweder vor oder nach einem Befehl WEB RECEIVE verwenden, mit dem der Nachrichtentext empfangen wird. Wenn Sie CICS als HTTP-Client einsetzen, müssen Sie zunächst die Nachricht mit dem Befehl WEB RECEIVE empfangen; anschließend können Sie die Header mit dem Befehl WEB READ HTTPHEADER lesen.

Im Abschnitt [HTTP-Headerreferenz für CICS-Webunterstützung](#) sind HTTP/1.1-Header aufgeführt, die Sie wahrscheinlich empfangen werden; dort finden Sie außerdem Anleitungen für die Maßnahmen, die Sie möglicherweise als Reaktion auf die Header ergreifen müssen.

Mit den Suchbefehlen für HTTP-Header (WEB STARTBROWSE HTTPHEADER, WEB READNEXT HTTPHEADER, WEB ENDBROWSE HTTPHEADER) können alle HTTP-Headerinformationen für eine Nachricht durchsucht werden.

Optionen

HTTPHEADER(datenbereich)

Gibt den Namen des HTTP-Headers an, der extrahiert werden soll.

NAMELENGTH(datenwert)

Gibt als Vollwort-Binärwert die Länge des Namens für den HTTP-Header an.

SESTOKEN(datenwert)

Wenn CICS als HTTP-Client eingesetzt wird, ist diese Option erforderlich. Sie gibt den 8 Byte umfassenden Binärwert für das Sitzungstoken an, das eine Verbindung zwischen CICS und einem Server eindeutig kennzeichnet. Dieser Wert wird durch einen Befehl WEB OPEN für CICS als HTTP-Client zurückgegeben. Die Verwendung des Sitzungstokens ist im Abschnitt [Sitzungstokens](#) erläutert.

VALUE(datenbereich)

Gibt den Puffer an, der den Wert des extrahierten HTTP-Headers aufnehmen soll.

VALUELENGTH(datenbereich)

Gibt als Vollwort-Binärzahlvariable die Länge des in der Option VALUE angegebenen Puffers an und wird auf die tatsächliche Länge der Daten gesetzt, die an die Anwendung zurückgegeben werden. Falls die Daten die Puffergröße überschreiten, tritt eine Bedingung LENGERR auf und die Daten werden abgeschnitten.

Bedingungen

16 INVREQ

Die RESP2-Werte lauten wie folgt:

1

Der Befehl wurde in einer Anwendung ausgegeben, die nicht zur CICS-Webunterstützung gehört.

3

Der Befehl wurde für eine Anforderung ausgegeben, die keine HTTP-Anforderung ist.

43

Es wurden keine HTTP-Header gefunden.

22 LENGERR

Die RESP2-Werte lauten wie folgt:

1

Der Wert in der Option VALUELENGTH ist nicht größer als Null (bei Verwendung von CICS als HTTP-Server).

2

Der Headerwert wurde abgeschnitten, weil der Empfangspuffer zu klein ist (bei Verwendung von CICS als HTTP-Server).

35

Der Wert in der Option NAMELENGTH ist nicht größer als Null (bei Verwendung von CICS als HTTP-Client).

52

Der Headerwert wurde abgeschnitten, weil der Empfangspuffer zu klein ist (bei Verwendung von CICS als HTTP-Client).

55

Der Wert in der Option VALUELENGTH ist nicht größer als Null (bei Verwendung von CICS als HTTP-Client).

13 NOTFND

Der RESP2-Wert lautet wie folgt:

1

Der Header mit dem angegebenen Namen war nicht zu finden.

19 NOTOPEN

Der RESP2-Wert lautet wie folgt:

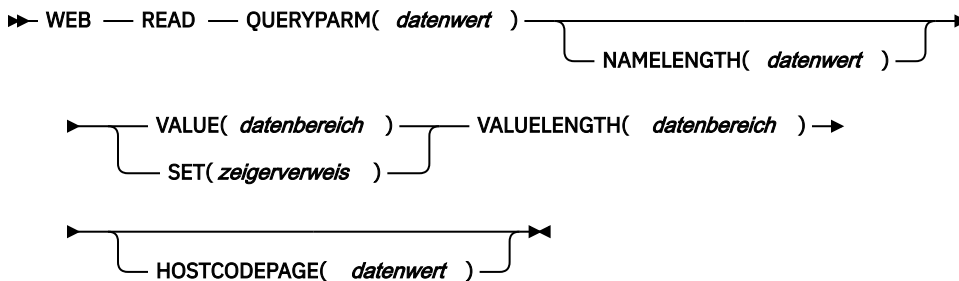
27

Das Sitzungstoken ist ungültig.

WEB READ QUERYPARAM

Liest Name/Wert-Paare aus einer Abfragezeichenfolge in einer URL.

WEB READ QUERYPARAM



Bedingungen: INVREQ, LENGERR, NOTFND

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl `WEB READ QUERYPARM` liest einen Schlüsselwortparameter, der aus einem Name/Wert-Paar besteht, aus einer Abfragezeichenfolge in einer URL und gibt ihn in einer angegebenen Codepage zurück. Zeichen mit Escapezeichen in der Abfragezeichenfolge werden in der Codepage ohne Escapezeichen dargestellt.

Der Befehl WEB READ QUERYPARM verarbeitet Abfragezeichenfolgedaten für alle HTTP-Methoden inklusive GET, POST, PUT und DELETE. Für Formulare (also Nachrichten mit den Medientypen 'application/x-www-form-urlencoded' oder 'multipart/form-data') können Sie weiterhin den vorhandenen Befehl WEB READ FORMFIELD verwenden. Obwohl der Befehl WEB READ FORMFIELD Name/Wert-Paare aus einer Abfragezeichenfolge lesen kann, führt er dies nur aus, wenn die HTTP-Methode GET verwendet wird, weil er davon ausgeht, dass es sich bei der Nachricht um ein HTML-Formular handelt.

Mit den Suchbefehlen für Abfragezeichenfolgen (WEB STARTBROWSE QUERYPARM, WEB READNEXT QUERYPARM und WEB ENDBROWSE QUERYPARM) können Sie alle Abfrageparameter in einer URL durchsuchen.

CICS liest Abfragezeichenfolgedaten nur dann, wenn CICS die Rolle des HTTP-Servers innehat. Beim Einsatz von CICS als HTTP-Client ist die Funktion nicht verfügbar.

Optionen

QUERYPARM(datenwert)

Gibt den Namen des Abfrageparameters an, der extrahiert werden soll. Geben Sie eine Textzeichenfolge an, die den Namen des angeforderten Parameters enthält. Die Groß-/Kleinschreibung wird bei der Textzeichenfolge nicht beachtet. Falls Sie die Option HOSTCODEPAGE angeben, müssen Sie den Namen des Abfrageparameters in der Codepage angeben, die Sie für diese Option festlegen.

HOSTCODEPAGE(datenwert)

Gibt den 8 Zeichen langen Namen der CICS-Codepage (für den Host) an, die durch das Anwendungsprogramm erforderlich ist. Normalerweise handelt es sich hierbei um eine EBCDIC-Codepage. CICS konvertiert den Wert des Abfrageparameters in diese Codepage, bevor er ihn als Wert der Option VALUE zurückgibt.

Das CICS-Standardformat für einen Codepagenamen besteht aus der Codepagenummer (oder allgemeiner der CCSID), die nach Bedarf mithilfe von 3 bis 5 Dezimalziffern geschrieben und anschließend mit nachgestellten Leerzeichen auf 8 Zeichen aufgefüllt wird. Bei der Codepage 37, die aus weniger als 3 Ziffern besteht, lautet das Standardformat '037'. CICS akzeptiert als Codepagenamen auch eine beliebige Dezimalzahl mit bis zu 8 Ziffern (aufgefüllt mit nachgestellten Leerzeichen) im Bereich von 1 bis 65535, auch wenn dieser nicht im Standardformat angegeben ist.

Falls die Codepage nicht angegeben ist, werden die Daten in der EBCDIC-Codepage zurückgegeben, die im Systeminitialisierungsparameter LOCALCCSID angegeben ist (der für die lokale CICS-Region gilt und den Standardwert 037 besitzt), unter der Voraussetzung, dass die angegebene Codepage durch die CICS-Webschnittstelle unterstützt wird. Die Codepage wird unterstützt, wenn sie in einer Liste von EBCDIC-Codepages angegeben ist, die von CICS als ausreichender Standard für eine erfolgreiche Analyse von Web-Headern anerkannt werden (hierzu zählen alle SBCS-CECP und Euro-Codepages). Andernfalls gibt CICS die Daten stattdessen in der EBCDIC-Standardcodepage 037 zurück.

NAMELENGTH(datenwert)

Gibt als Vollwort-Binärwert die Länge des Abfrageparameternamens an.

SET(zeigerverweis)

Gibt den Zeigerverweis an, der auf die Adresse der empfangenen Daten gesetzt werden soll.

VALUE(datenbereich)

Gibt den Puffer an, der den Wert des benannten Abfrageparameters aufnehmen soll. CICS entfernt alle etwaigen Escapezeichen bei Zeichen mit Escapezeichen, bevor sie in den Puffer gestellt werden.

VALUELENGTH(datenbereich)

Gibt als Vollwort-Binärwert die Länge des Abfrageparameterwertes an. In diesem Datenbereich wird die tatsächliche Länge des Wertes zurückgegeben. Falls Sie die Option VALUE angeben, gibt die Option VALUELENGTH die maximale Länge der Daten an, die das Programm akzeptiert. Wenn der Wert die Länge des Puffers überschreitet, wird er abgeschnitten. Ist die Länge des Abfrageparameterwertes kleiner als die Größe des Puffers, wird der Abfrageparameterwert an die äußeren linken Bytepositionen gestellt.

Bedingungen

16 INVREQ

Die RESP2-Werte lauten wie folgt:

1

Der Befehl wurde in einer Anwendung ausgegeben, die nicht zur CICS-Webunterstützung gehört.

3

Der Befehl wurde für eine Anforderung ausgegeben, die keine HTTP-Anforderung ist.

12

Die Host-Codepage wurde nicht gefunden.

13

In der HTTP-Anforderung wurden keine Schlüsselwortparameter übergeben.

14

Die Kombination der Codepages für den Client und den Server ist ungültig.

17

In der HTTP-Anforderung wurden ungültige Schlüsselwortparameter gefunden.

22 LENGERR

Die RESP2-Werte lauten wie folgt:

1

Die Länge des Wertes in der Option NAMELENGTH oder VALUELENGTH ist kleiner-gleich Null.

5

Der Schlüsselwortparameterwert wurde während einer Leseoperation abgeschnitten, weil der Empfangspuffer zu klein ist.

13 NOTFND

Der RESP2-Wert lautet wie folgt:

1

Der Schlüsselwortparameter mit dem angegebenen Namen war nicht zu finden.

WEB READNEXT FORMFIELD

Ruft das nächste Name/Wert-Paar in einem HTML-Formular ab.

WEB READNEXT FORMFIELD

► WEB — READNEXT — FORMFIELD(*datenbereich*) — NAMELENGTH(*datenbereich*) →
 ► VALUE(*datenbereich*) — VALUELENGTH(*datenbereich*) ◀

Bedingungen: ENDFILE, INVREQ, LENGERR

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl **WEB READNEXT FORMFIELD** ruft das nächste Name/Wert-Paar in einem HTML-Formular ab.

Die Daten werden in ihrem Format ohne Escapezeichen zurückgegeben (eine Erläuterung finden Sie unter [Reservierte und ausgeschlossene Zeichen](#)).

Optionen

FORMFIELD(*datenbereich*)

Gibt den Puffer an, der den Namen des abgerufenen Formularfeldes aufnehmen soll. Die Groß-/Kleinschreibung des Namens entspricht der des im Formular gespeicherten Wertes.

NAMELENGTH(*datenbereich*)

Gibt als Vollwort-Binärwert die Länge des Formularfeldnamens an. In diesem Datenbereich wird die tatsächliche Länge des Namens zurückgegeben.

Ist die Länge des Formularfeldnamens kleiner als die Größe des Puffers, wird der Formularfeldname an die äußeren linken Bytepositionen gestellt.

Falls der Formularfeldname die Puffergröße überschreitet, tritt eine Bedingung LENGERR auf und der Name wird abgeschnitten. Bei der nächsten Ausführung von **WEB READNEXT FORMFIELD** wird das Feld, das den Fehler verursacht hat, erneut zurückgegeben. Verwenden Sie einen Puffer, der groß genug ist, um den Namen aufzunehmen.

VALUE(*datenbereich*)

Gibt den Puffer an, von dem der Wert des aufgenommen werden soll, der dem im Datenbereich der Option FROMFIELD zurückgegebenen Namen entspricht. CICS entfernt alle etwaigen Escapezeichen bei Zeichen mit Escapezeichen, bevor sie in den Puffer gestellt werden.

VALUELENGTH(*datenbereich*)

Gibt als Vollwort-Binärwert die Länge des Formularfeldwertes an. In diesem Datenbereich wird die tatsächliche Länge des Wertes zurückgegeben.

Ist die Länge des Formularfeldwertes kleiner als die Größe des Puffers, wird der Formularfeldwert an die äußeren linken Bytepositionen gestellt.

Falls der Wert größer ist als der Puffer, tritt eine Bedingung LENGERR auf und der Wert wird abgeschnitten. Bei der nächsten Ausführung von **WEB READNEXT FORMFIELD** wird das Feld, das den Fehler verursacht hat, erneut zurückgegeben. Verwenden Sie einen Puffer, der groß genug ist, um den Wert aufzunehmen.

Bedingungen

20 ENDFILE

Das Ende der Liste von Name/Wert-Paaren wurde erreicht.

16 INVREQ

Die RESP2-Werte lauten wie folgt:

1

Der Befehl wurde in einer Anwendung ausgegeben, die nicht zur CICS-Webunterstützung gehört.

3

Der Befehl wurde für eine Anforderung ausgegeben, die keine HTTP-Anforderung ist.

4

Der Befehl wurde vor Ausgabe eines Befehls **WEB STARTBROWSE FORMFIELD** ausgegeben.

6

Es wurde ein Formularfeld gefunden, das nicht das Format NAME:WERT besitzt.

153

Der Formulartyp ist unbekannt.

154

In den Formulardaten wurde eine Begrenzungszeichenfolge erwartet, jedoch nicht gefunden.

22 LENGERR

Die RESP2-Werte lauten wie folgt:

1

Der Wert in der Option NAMELENGTH oder VALUELENGTH ist kleiner-gleich Null.

4

Der Formularfeldname wurde während eines Suchvorgangs abgeschnitten, weil der Empfangspuffer zu klein ist.

Bei der nächsten Ausführung von **WEB READNEXT FORMFIELD** wird das Feld, das den Fehler verursacht hat, erneut zurückgegeben. Das heißt, dass die Bedingung LENGERR bestehen bleibt, bis ein Puffer verwendet wird, der groß genug für den Namen ist.

5

Der Formularfeldwert wurde abgeschnitten, weil der Empfangspuffer zu klein ist.

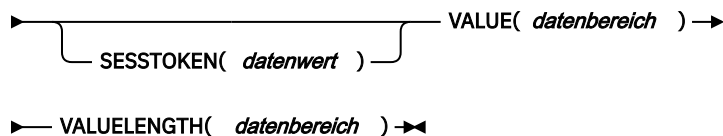
Bei der nächsten Ausführung von **WEB READNEXT FORMFIELD** wird das Feld, das den Fehler verursacht hat, erneut zurückgegeben. Das heißt, dass die Bedingung LENGERR bestehen bleibt, bis ein Puffer verwendet wird, der groß genug für den Wert ist.

WEB READNEXT HTTPHEADER

Ruft den nächsten HTTP-Header ab.

WEB READNEXT HTTPHEADER

➤ WEB — READNEXT — HTTPHEADER(*datenbereich*) — NAMELENGTH(*datenbereich*) →



Bedingungen: ENDFILE, INVREQ, LENGERR, NOTOPEN

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl **WEB READNEXT HTTPHEADER** ruft den nächsten HTTP-Header in der Liste der Header ab. Die Option **SESSTOKEN** ist erforderlich, falls die HTTP-Headerinformationen Teil einer Antwort sind, die an CICS als HTTP-Client gesendet wird.

Optionen

HTTPHEADER(*datenbereich*)

Gibt den Puffer an, der den Namen des extrahierten HTTP-Headers aufnehmen soll.

NAMELENGTH(*datenbereich*)

Gibt als Vollwort-Binärdatenbereich die Länge des in der Option **HTTPHEADER** angegebenen Puffers an und wird auf die tatsächliche Länge der Daten gesetzt, die an die Anwendung zurückgegeben werden. Falls die Daten die Puffergröße überschreiten, tritt eine Bedingung **LENGERR** auf und die Daten werden abgeschnitten.

SESSTOKEN(*datenwert*)

Wenn CICS als HTTP-Client eingesetzt wird, ist diese Option erforderlich. Sie gibt den 8 Byte umfassenden Binärwert für das Sitzungstoken an, das eine Verbindung zwischen CICS und einem Server eindeutig kennzeichnet. Dieser Wert wird durch einen Befehl **WEB OPEN** für CICS als HTTP-Client zurückgegeben. Die Verwendung des Sitzungstokens ist im Abschnitt [Sitzungstokens](#) erläutert.

VALUE(*datenbereich*)

Gibt den Puffer an, der den Wert des extrahierten HTTP-Headers aufnehmen soll.

VALUELENGTH(*datenbereich*)

Gibt als Vollwort-Binärdatenbereich die Länge des in der Option **VALUE** angegebenen Puffers an und wird auf die tatsächliche Länge der Daten gesetzt, die an die Anwendung zurückgegeben werden. Falls die Daten die Puffergröße überschreiten, tritt eine Bedingung **LENGERR** auf und die Daten werden abgeschnitten.

Bedingungen

20 ENDFILE

Das Ende der Liste von HTTP-Headern wurde erreicht.

16 INVREQ

Die RESP2-Werte lauten wie folgt:

- 1 Der Befehl wurde in einer Anwendung ausgegeben, die nicht zur CICS-Webunterstützung gehört.
- 3 Der Befehl wurde für eine Anforderung ausgegeben, die keine HTTP-Anforderung ist.
- 4 Der Befehl wurde vor Ausgabe eines Befehls **WEB STARTBROWSE** ausgegeben.
- 6 Es wurde ein Header gefunden, der nicht das Format **NAME:WERT** besitzt.

22 LENGERR

Die RESP2-Werte lauten wie folgt:

1

Die Länge des Wertes in der Option NAMELENGTH oder VALUELENGTH ist kleiner-gleich Null (bei Verwendung von CICS als HTTP-Server).

4

Der Headername wurde abgeschnitten, weil der Empfangspuffer zu klein ist (bei Verwendung von CICS als HTTP-Server).

5

Der Headerwert wurde abgeschnitten, weil der Empfangspuffer zu klein ist (bei Verwendung von CICS als HTTP-Server).

35

Die Länge des Wertes in der Option NAMELENGTH ist kleiner-gleich Null (bei Verwendung von CICS als HTTP-Client).

51

Der Headername wurde abgeschnitten, weil der Empfangspuffer zu klein ist (bei Verwendung von CICS als HTTP-Client).

52

Der Headerwert wurde abgeschnitten, weil der Empfangspuffer zu klein ist (bei Verwendung von CICS als HTTP-Client).

55

Die Länge des Wertes in der Option VALUELENGTH ist kleiner-gleich Null (bei Verwendung von CICS als HTTP-Client).

19 NOTOPEN

Der RESP2-Wert lautet wie folgt:

27

Das Sitzungstoken ist ungültig.

WEB READNEXT QUERYPARM

Ruft das nächste Name/Wert-Paar in den Abfragezeichenfolgedaten einer URL ab.

WEB READNEXT QUERYPARM

➡ WEB — READNEXT — QUERYPARM(*datenbereich*) — NAMELENGTH(*datenbereich*) →
 ➡ VALUE(*datenbereich*) — VALUELENGTH(*datenbereich*) →

Bedingungen: ENDFILE, INVREQ, LENGERR

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl WEB READNEXT QUERYPARM ruft den nächsten Schlüsselwortparameter (Name/Wert-Paar) in einer Abfragezeichenfolge aus einer URL ab.

Die Daten werden in ihrem Format ohne Escapezeichen zurückgegeben (eine Erläuterung finden Sie unter Reservierte und ausgeschlossene Zeichen).

Optionen

QUERYPARM(*datenbereich*)

Gibt den Puffer an, der den Namen des abgerufenen Schlüsselwortparameters aufnehmen soll. Die Groß-/Kleinschreibung des Namens entspricht der des Schlüsselwortparameter gespeicherten Wertes.

NAMELENGTH(datenbereich)

Gibt als Vollwort-Binärwert die Länge des Schlüsselwortparameternamens an. In diesem Datenbereich wird die tatsächliche Länge des Namens zurückgegeben. Ist die Länge des Schlüsselwortparameternamens kleiner als die Größe des Puffers, wird der Schlüsselwortparametername an die äußeren linken Bytepositionen gestellt.

VALUE(datenbereich)

Gibt den Puffer an, von dem der Wert des aufgenommen werden soll, der dem im Datenbereich der Option QUERYPARM zurückgegebenen Namen entspricht. CICS entfernt alle etwaigen Escapezeichen bei Zeichen mit Escapezeichen, bevor sie in den Puffer gestellt werden.

VALUELENGTH(datenbereich)

Gibt als Vollwort-Binärwert die Länge des Schlüsselwortparameterwertes an. In diesem Datenbereich wird die tatsächliche Länge des Wertes zurückgegeben. Wenn der Wert die Länge des Puffers überschreitet, wird er abgeschnitten. Ist die Länge des Schlüsselwortparameterwertes kleiner als die Größe des Puffers, wird der Schlüsselwortparameterwert an die äußeren linken Bytepositionen gestellt.

Bedingungen**20 ENDFILE**

Das Ende der Liste von Schlüsselwortparametern wurde erreicht.

16 INVREQ

Die RESP2-Werte lauten wie folgt:

- 1**
Der Befehl wurde in einer Anwendung ausgegeben, die nicht zur CICS-Webunterstützung gehört.
- 3**
Der Befehl wurde für eine Anforderung ausgegeben, die keine HTTP-Anforderung ist.
- 4**
Der Befehl wurde vor Ausgabe eines Befehls WEB STARTBROWSE ausgegeben.
- 6**
Es wurde ein Schlüsselwortparameter gefunden, der nicht das Format NAME=WERT besitzt.

22 LENGERR

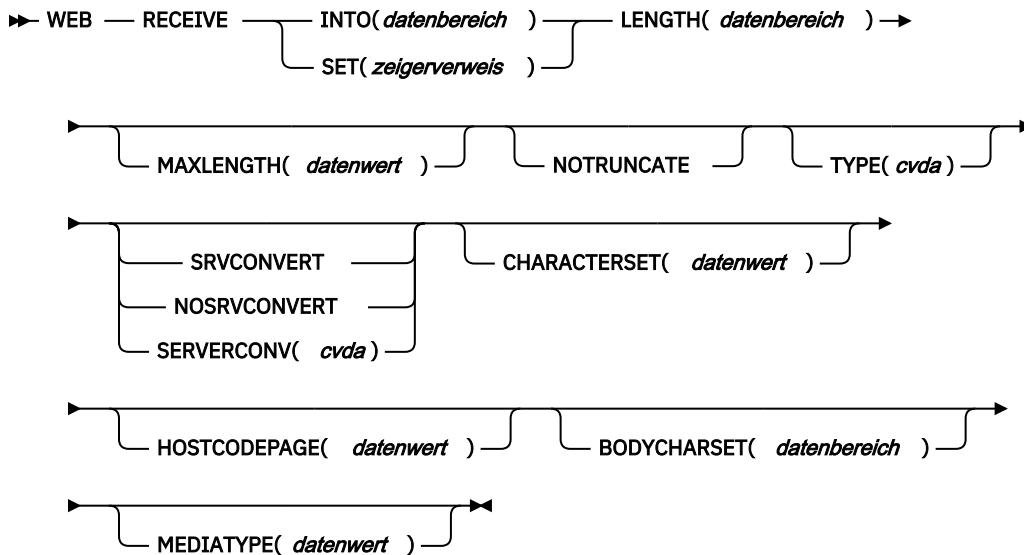
Die RESP2-Werte lauten wie folgt:

- 1**
Der Wert in der Option NAMELENGTH oder VALUELENGTH ist kleiner-gleich Null.
- 4**
Der Schlüsselwortparametername wurde während eines Suchvorgangs abgeschnitten, weil der Empfangspuffer zu klein ist.
- 5**
Der Schlüsselwortparameterwert wurde abgeschnitten, weil der Empfangspuffer zu klein ist.

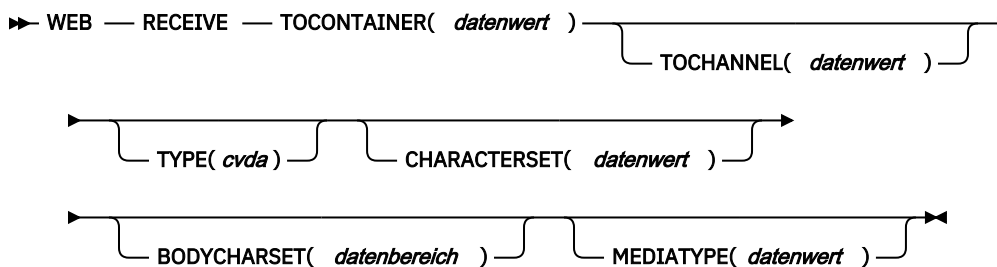
WEB RECEIVE (Server)

Empfängt eine HTTP-Anforderung oder eine Nicht-HTTP-Nachricht.

WEB RECEIVE (CICS als HTTP-Server mit Pufferverwendung)



WEB RECEIVE (CICS als HTTP-Server mit Containerverwendung)



Bedingungen: CHANNELERR, CONTAINERERR, INVREQ, LENGERR, NOTFND

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl WEB RECEIVE empfängt den Hauptteil einer HTTP-Anforderung (bzw. bei einer Nicht-HTTP-Nachricht alle Daten) in einem von der Anwendung bereitgestellten oder einem festgelegten Puffer. Alternativ kann eine HTTP-Anforderung in einem benannten Container gespeichert werden. Die Header für eine HTTP-Anforderung können mit den Befehlen WEB HTTPHEADER separat geprüft werden. Gegenstand des Befehls WEB RECEIVE kann Folgendes sein:

- Der Hauptteil einer HTTP-Anforderung, die von einem Web-Client an CICS als HTTP-Server ausgegeben wurde. Anleitungen für die korrekte Verwendung des Befehls WEB RECEIVE für diesen Zweck enthält der Abschnitt Entitätshauptteil einer HTTP-Anforderung empfangen.
- Eine von den CICS-Webunterstützungsfunktionen verarbeitete Nicht-HTTP-Nachricht, bei der in der TCIPSERVICE-Definition das benutzerdefinierte Protokoll (USER) angegeben ist. Anweisungen für Nicht-HTTP-Nachrichten finden Sie unter [CICS-Webunterstützung und Nicht-HTTP-Anforderungen](#).
- Eine Anforderung aus einer anderen Anwendung, die unter Verwendung der CICS-Geschäftslogik-schnittstelle direkt Kontakt zum Anwendungsprogramm aufgenommen hat, statt hierzu den HTTP-Listener von CICS einzusetzen. Anweisungen für die CICS-Geschäftslogik-schnittstelle enthält der Abschnitt [Einführung in die CICS-Geschäftslogik-schnittstelle](#).

Die Daten werden in ihrem Format mit Escapezeichen zurückgegeben.

Wenn ein HTTP-Hauptteil in einem Anwendungspuffer empfangen wird (mithilfe der Optionen INTO oder SET), besteht beim Befehl **WEB RECEIVE** die Möglichkeit, den Typ der Codepagekonvertierung für eingehende Daten anzugeben, die durch das CICS-Anwendungsprogramm empfangen werden. Falls Sie keine

der Optionen für die Codepagekonvertierung angeben (SERVERCONV, CLNTCODEPAGE, CHARACTERSET, HOSTCODEPAGE), findet keine Codepagekonvertierung statt.

Beim Empfang eines HTTP-Hauptteils in einem benannten Container ist die Codepagekonvertierung nicht zulässig. Falls der benannte Container durch die Benutzeranwendung vor Ausgabe des Befehls **WEB RECEIVE** erstellt wurde, wird der Container gelöscht und erneut erstellt. Die Informationen zum Medientyp im Header 'Content-Type' der HTTP-Anforderung bestimmen, ob der Container mit dem Datentyp BIT oder CHAR erneut erstellt wird (es sei denn, die Option CHARACTERSET war angegeben; in diesem Fall geht CICS davon aus, dass die im Container gespeicherten Daten in der Codepage codiert sind, die in der Option CHARACTERSET angegeben ist).

- Falls der Medientyp ein Textmedientyp ist, wird ein Container mit dem Datentyp CHAR erstellt.
- Ist der Medientyp ein nicht für Text bestimmter Medientyp, wird ein Container mit dem Datentyp BIT erstellt.
- Falls die HTTP-Anforderung keine Informationen zum Medientyp enthält, wird standardmäßig vom Textmedientyp ausgegangen.

Falls ein Container mit dem Datentyp CHAR erstellt wird, wird der bei IANA registrierte Name für die ID des codierten Zeichensatzes (Coded Character Set Identifier, CCSID) für die aktuelle Codepage der Daten aus dem Element 'charset' im Header 'Content-Type' der HTTP-Anforderung abgerufen. Falls diese Informationen nicht bereitgestellt oder nicht von CICS unterstützt werden, wird als Standardwert ISO-8859-1 angenommen.

Der Zeichensatz kann mithilfe der Option CHARACTERSET überschrieben werden. Bei Angabe der Option CHARACTERSET wird ein Container mit dem Datentyp CHAR erstellt.

Die Optionen LENGTH, MAXLENGTH, NOTRUNCATE, SERVERCONV und HOSTCODEPAGE sind nicht zulässig, wenn ein HTTP-Hauptteil in einem benannten Container empfangen wird.

Für den Empfang von Nachrichten, die über einen Benutzerprotokollsocket gesendet werden, können keine Container verwendet werden.

CICS empfängt den Hauptteil der HTTP-Serveranforderung im 64-Bit-Speicher (oberhalb der 2-GB-Grenze), falls die HTTP-Anforderung einen Header 'Content-Length' enthält oder beim Senden in Blöcke aufgeteilt wird. Der Hauptteil wird nur dann in den 31-Bit-Speicher (oberhalb der 16-MB-Grenze) kopiert, wenn CICS eine 31-Bit-Version für die Serveranwendung bereitstellen muss.

Optionen

BODYCHARSET(datenbereich)

Gibt den Zeichensatz des HTTP-HTTP-Anforderungshauptteils an.

Der Name des Zeichensatzes kann aus bis zu 40 alphanumerischen Zeichen inklusive der entsprechenden Interpunktion bestehen.

Wenn der HTTP-Hauptteil in einem Anwendungspuffer empfangen wird, wird der Zeichensatz wie folgt zurückgegeben:

- Falls die Option INTO oder SET angegeben ist und der HTTP-Hauptteil konvertiert wird, gibt CICS den Zeichensatz des HTTP-Hauptteils vor der Konvertierung zurück.
- Falls die Option INTO oder SET angegeben ist und der HTTP-Hauptteil nicht konvertiert wird, gibt CICS den Zeichensatz zurück, der im Header 'Content-Type' angegeben ist. Wenn keine Zeichensatzinformationen verfügbar sind, werden Leerzeichen zurückgegeben.

Wenn der HTTP-Hauptteil in einem benannten Container empfangen wird, wird der Zeichensatz wie folgt zurückgegeben:

- Falls der Container den Datentyp CHAR besitzt, gibt CICS den Zeichensatz der codierten Daten zurück.
- Falls der Container den Datentyp BIT besitzt, gibt CICS Leerzeichen zurück.

Wenn der zurückgegebene Wert mehr als 40 Byte umfasst, werden die Daten abgeschnitten. Umfasst der zurückgegebene Wert weniger als 40 Byte, werden die Daten rechts mit Leerzeichen aufgefüllt.

CHARACTERSET(datenwert)

Gibt den Zeichensatz an, den der Web-Client für den Entitätshauptteil des empfangenen Elements verwendet hat. Der Name des Zeichensatzes kann aus bis zu 40 alphanumerischen Zeichen inklusive der entsprechenden Interpunktion bestehen. CICS unterstützt nicht alle von IANA genannten Zeichensätze. Im Abschnitt HTML-codierte Zeichensätze sind die IANA-Zeichensätze aufgelistet, die durch CICS für die Codepagekonvertierung unterstützt werden.

Falls der HTTP-Anforderungshauptteil in einem Puffer gespeichert wird und die Option CHARACTERSET angegeben ist, wird die Option SRVCONVERT ebenfalls angenommen, weshalb eine Codepagekonvertierung für den Entitätshauptteil stattfindet. CICS kann den Zeichensatz für den Nachrichtenhauptteil beim Empfang von Daten in einem Puffer (unter Verwendung der Option INTO oder SET) ermitteln, wenn Sie die Option SRVCONVERT und/oder HOSTCODEPAGE (ohne Angabe der Option CHARACTERSET) angeben. Die Beschreibung für die Option SERVERCONV enthält Anweisungen für einen solchen Fall.

Falls der HTTP-Anforderungshauptteil in einem Container gespeichert wird und die Option CHARACTERSET angegeben ist, geht CICS davon aus, dass die im Container gespeicherten Daten in dieser Codepage codiert sind. Der Wert von CHARACTERSET überschreibt den Zeichensatz im Header 'Content-Type' der empfangenen Daten und legt die CCSID des Containers auf den für CHARACTERSET angegebenen Wert fest. Dies bedeutet, dass bei Ausgabe des Befehls GET CONTAINER mit diesem Container die Daten aus der CCSID, die durch den Parameter CHARACTERSET im Befehl WEB RECEIVE festgelegt wurde, in eine beliebige andere, vom Benutzer angeforderte Codepage konvertiert werden.

CLNTCODEPAGE(datenwert)

Diese Option wird nur zu Upgradezwecken unterstützt. Sie wurde durch die Option CHARACTERSET ersetzt. Bei beiden Schlüsselwörtern führt CICS dieselbe Aktion aus.

HOSTCODEPAGE(datenwert)

Gibt den 8 Zeichen langen Namen der CICS-Codepage (für den Host) an, die durch das Anwendungsprogramm verwendet wird und in die der Entitätshauptteil des empfangenen Elements aus dem Zeichensatz, in dem er vom Web-Client empfangen wurde, konvertiert werden soll.

Das CICS-Standardformat für einen Codepagenamen besteht aus der Codepagenummer (oder allgemeiner der CCSID), die nach Bedarf mithilfe von 3 bis 5 Dezimalziffern geschrieben und anschließend mit nachgestellten Leerzeichen auf 8 Zeichen aufgefüllt wird. Bei der Codepage 37, die aus weniger als 3 Ziffern besteht, lautet das Standardformat '037'. CICS akzeptiert als Codepagenamen auch eine beliebige Dezimalzahl mit bis zu 8 Ziffern (aufgefüllt mit nachgestellten Leerzeichen) im Bereich von 1 bis 65535, auch wenn dieser nicht im Standardformat angegeben ist.

Wenn Sie Daten in einem Puffer empfangen (und eine der Optionen INTO oder SET angegeben ist), ist die Option HOSTCODEPAGE angegeben und die Option SRVCONVERT wird angenommen, weshalb eine Codepagekonvertierung für den Entitätshauptteil stattfindet. Wenn Sie die Option SRVCONVERT und/oder CHARACTERSET angeben, die Option HOSTCODEPAGE jedoch nicht, wird CICS die Ermittlung der Host-Codepage überlassen.

Wenn diese Option nicht angegeben ist, ist der Standardwert die Standardcodepage für die lokale CICS-Region, die im Systeminitialisierungsparameter LOCALCCSID angegeben war.

Geben Sie die Option HOSTCODEPAGE nicht an, falls Sie die Option TOCONTAINER verwenden.

INTO(datenbereich)

Gibt den Puffer an, der die empfangenen Daten enthalten soll. Falls die HTTP-Anforderung einen Header 'Content-Length' enthält oder in Blöcken gesendet wurde, bewahrt CICS den HTTP-Hauptteil im internen 64-Bit-Speicher auf; andernfalls bewahrt CICS den Hauptteil im internen 31-Bit-Speicher auf. Der Hauptteil wird aus diesem internen Speicher in den Anwendungspuffer kopiert, der in der Option **INTO** angegeben ist.

Falls die Option **INTO** angegeben ist, muss auch die Option **MAXLENGTH** mit einem größeren Wert als Null angegeben sein (andernfalls wird an die Anwendung ein Fehler INVREQ mit dem RESP2-Wert 16 zurückgegeben).

LENGTH(*datenbereich*)

Gibt eine Vollwort-Binärzahlvariable an, die auf die Menge der Daten gesetzt wird, die CICS an die Anwendung zurückgegeben hat. Dieser Wert kann - insbesondere bei Einbeziehung eines Doppelbyte- oder Mehrbytezeichensatzes - etwas niedriger als der mit der Option MAXLENGTH festgelegte Grenzwert sein, weil CICS am Ende der Daten kein abgeschnittenes Zeichen zurückgibt.

- Falls die Option NOTRUNCATE **nicht angegeben ist**, sind alle weiteren Daten in der Nachricht jetzt gelöscht worden. Falls weitere Daten vorhanden sind, wird eine Antwort LENGERR mit dem RESP2-Wert 57 zurückgegeben.
- Falls die Option NOTRUNCATE **angegeben ist**, werden alle zusätzlichen Daten aufbewahrt. Falls zusätzliche Daten vorhanden sind, wird eine Antwort LENGERR mit dem RESP2-Wert 36 zurückgegeben. Die Beschreibung für die Option NOTRUNCATE enthält Anweisungen für einen solchen Fall.

Falls Sie zum Speichern des HTTP-Hauptteils einen Anwendungspuffer verwenden, muss die Option LENGTH angegeben sein, wenn die Option INTO oder SET verwendet wird. Geben Sie die Option LENGTH bei Verwendung eines benannten Containers zum Speichern des HTTP-Hauptteils (und somit bei Angabe der Option TOCONTAINER) nicht an.

MAXLENGTH(*datenwert*)

Gibt als Vollwort-Binärwert die maximale Datenmenge an, die CICS an die Anwendung übergeben soll. Die Option MAXLENGTH gilt unabhängig davon, ob die Option INTO oder SET für den Empfang von Daten angegeben ist. Falls die Daten unter Verwendung von Chunked Transfer Coding gesendet wurden, assembliert CICS die Blöcke in einer einzigen Nachricht, bevor diese an die Anwendung übergeben wird. Die Option MAXLENGTH gilt daher für die Gesamtlänge der in Blöcke aufgeteilten Nachricht und nicht für jeden einzelnen Block. Die Messung der Daten findet nach einer gegebenenfalls erfolgten Codepagekonvertierung statt. Falls die Länge der Daten den angegebenen Wert überschreitet und die Option NOTRUNCATE **nicht angegeben ist**, werden die Daten auf diesen Wert abgeschnitten und die übrigen Daten werden gelöscht. Falls die Länge der Daten den angegebenen Wert überschreitet und die Option NOTRUNCATE **angegeben ist**, bewahrt CICS die verbleibenden Daten auf und verwendet sie zur Erfüllung nachfolgender Befehle RECEIVE.

Geben Sie die Option MAXLENGTH nicht an, falls Sie die Option TOCONTAINER verwenden.

MEDIATYPE (*datenbereich*)

Gibt den Dateninhalt eines bereitgestellten Nachrichtenhauptteils an, z. B. text/xml. Als Wert für die Option MEDIATYPE müssen Sie einen 56 Byte umfassenden Bereich angeben. Der Medientyp kann bis zu 56 alphanumerische Zeichen lang sein, inklusive der entsprechenden Interpunktion.

NOTRUNCATE

Gibt an, dass bei einer Überschreitung der in der Option MAXLENGTH angeforderten Länge durch die verfügbaren Daten die übrigen Daten nicht sofort gelöscht, sondern zum Abruf durch nachfolgende Befehle RECEIVE aufbewahrt werden sollen. (Falls keine weiteren Befehle RECEIVE ausgegeben werden, werden die Daten bei der Beendigung der Transaktion gelöscht.)

Ein einzelner Befehl RECEIVE mit der Option SET und ohne die Option MAXLENGTH empfängt alle verbleibenden Daten ungeachtet ihrer Länge. Alternativ können Sie eine Folge von Befehlen RECEIVE mit der Option NOTRUNCATE verwenden, um die verbleibenden Daten in geeigneten Blöcken zu empfangen. Setzen Sie den Befehl RECEIVE weiterhin so oft ab, bis Sie keine Antwort LENGERR mehr erhalten. Der Empfang einer kürzeren Länge, als mit der Option MAXLENGTH angefordert wird, nicht zwangsläufig ein Hinweis auf das Ende der Daten. Diese Situation könnte vorliegen, wenn CICS die Rückgabe eines abgeschnittenen Zeichens am Ende der Daten vermeiden muss.

Geben Sie die Option NOTRUNCATE nicht an, falls Sie die Option TOCONTAINER verwenden. Der gesamte HTTP-Hauptteil wird in dem Container gespeichert, der im ersten Befehl **WEB RECEIVE** angegeben war.

SERVERCONV(*cvda*)

Gibt an, ob CICS den Entitätshauptteil des empfangenen Elements aus dem vom Web-Client verwendeten Zeichensatz in eine für die Anwendung geeignete Codepage konvertiert oder nicht. Mit den Optionen CHARACTERSET und HOSTCODEPAGE können Sie in diesem Befehl den Zeichensatz und die Codepage angeben, die verwendet werden. Falls Sie eine dieser Optionen angeben, wird die Codepagekonvertierung (Option SRVCONVERT) angenommen. Alternativ können Sie eine oder beide dieser

Optionen weglassen, die Option SRVCONVERT angeben und CICS einen passenden Zeichensatz und eine geeignete Codepage festlegen lassen.

Geben Sie die Option SERVERCONV nicht an, falls Sie die Option TOCONTAINER verwenden.

SRVCONVERT

CICS konvertiert den Entitätshauptteil der Nachricht, bevor er an die Anwendung übergeben wird.

Wenn Sie die Option SRVCONVERT ohne die Option CHARACTERSET angeben, ermittelt CICS den Zeichensatz folgendermaßen:

1. Falls die Anforderung des Web-Clients einen Header 'Content-Type' enthält, der einen von CICS unterstützten Zeichensatz angibt, wird dieser Zeichensatz verwendet.
2. Falls die Anforderung des Web-Clients keinen Header 'Content-Type' enthält oder der benannte Zeichensatz nicht unterstützt wird, wird der Zeichensatz ISO-8859-1 verwendet.
3. Bei Nachrichten, die keine HTTP-Nachrichten sind (und mit dem für USER angegebenen Protokoll gesendet wurden), wird der Zeichensatz ISO-8859-1 verwendet.

Wenn Sie die Option SRVCONVERT ohne die Option HOSTCODEPAGE angeben, legt CICS als Host-Codepage die Standardcodepage für die lokale CICS-Region fest, die im Systeminitialisierungsparameter LOCALCCSID angegeben wurde.

Falls Sie die Option SRVCONVERT allein angeben, müssen Sie beachten, dass der Medientyp für die Nachricht einen Dateninhaltenstyp angeben muss, der gemäß den IANA-Definitionen als Text erkannt werden kann, damit eine Codepagekonvertierung stattfindet. Für Nachrichten, bei denen kein Medientyp, jedoch die Option SRVCONVERT angegeben ist, findet die Codepagekonvertierung ebenfalls statt. Falls ein Medientyp vorhanden ist, bei dem es sich nicht um Text handelt, konvertiert CICS den Nachrichtenhauptteil nicht. Zur Kompatibilität mit webfähigen Anwendungen, die in früheren Releases codiert wurden, hat der Medientyp für die Nachricht jedoch keinen Einfluss auf die Codepagekonvertierung, falls Sie eine der Optionen CHARACTERSET oder HOSTCODEPAGE angeben oder die Option SERVERCONV weglassen.

NOSRVCONVERT

CICS konvertiert nicht den Entitätshauptteil des Elements, der an die Anwendung in dem vom Web-Client verwendeten Zeichensatz übergeben wird. Falls Sie die Option NOSRVCONVERT angeben, ist eine Angabe der Optionen CHARACTERSET oder HOSTCODEPAGE nicht möglich.

SET(zeigerverweis)

Gibt den Zeigerverweis an, der auf die Adresse der empfangenen Daten gesetzt werden soll. Der Zeigerverweis ist bis zum nächsten Befehl RECEIVE oder bis zum Ende der Task gültig.

Falls die HTTP-Anforderung einen Header 'Content-Length' enthält oder in Blöcken gesendet wurde, bewahrt CICS den HTTP-Hauptteil im internen 64-Bit-Speicher auf; andernfalls bewahrt CICS den Hauptteil im internen 31-Bit-Speicher auf. Der Hauptteil wird aus diesem internen Speicher in den Speicher kopiert, der für die Anwendung zugänglich ist, und der Zeigerverweis wird auf diese Kopie gesetzt.

TOCHANNEL(datenwert)

Gibt den Namen des Kanals an, zu dem der Container gehört. Der Name des Kanals kann aus bis zu 16 alphanumerischen Zeichen inklusive der entsprechenden Interpunktion bestehen. Zulässige Zeichen: A-Z a-z 0-9 \$ @ # / % & ? ! : | " = , ; < > . - und _ . Führende und eingebettete Leerzeichen sind nicht zulässig. Falls der Name weniger als 16 Zeichen umfasst, wird er mit nachgestellten Leerzeichen aufgefüllt. Falls der Kanal nicht vorhanden ist, wird er erstellt. Dieser neue Kanal bleibt gültig, bis sich die Verbindungsebene ändert. Informationen zum Geltungsbereich von Kanälen finden Sie unter [Geltungsbereich eines Kanals](#).

Wenn Sie Ihre Kanäle zwischen CICS-Regionen verlagern wollen, denken Sie daran, dass Sie ausschließlich alphanumerische Zeichen (A-Z 0-9 & : = , ; < > . - _) verwenden; dies stellt sicher, dass die Zeichen von allen EBCDIC-Codepages identisch dargestellt werden.

Sie können den Kanalnamen DFHTRANSACTION angeben, um einen Transaktionskanal zu verwenden. Ein Transaktionskanal verliert nicht seine Gültigkeit, wenn sich die Verbindungsebene ändert; der Zu-

griff auf diesen Kanal ist in der Transaktion immer möglich. Weitere Informationen finden Sie unter Kanäle und Container.

Falls die Option TOCHANNEL nicht angegeben ist, geht CICS vom aktuellen Kanal aus.

TOCONTAINER(datenwert)

Gibt den Namen des Containers an, in den die Daten gestellt werden. Der Name des Containers kann aus bis zu 16 alphanumerischen Zeichen inklusive der entsprechenden Interpunktion bestehen. Zulässige Zeichen: A-Z a-z 0-9 \$ @ # / % & ? ! : | " = , ; > . - und _. Führende und eingebettete Leerzeichen sind nicht zulässig. Falls der Name weniger als 16 Zeichen umfasst, wird er mit nachgestellten Leerzeichen aufgefüllt.

Wenn Sie Ihre Container zwischen CICS-Regionen verlagern wollen, denken Sie daran, dass Sie ausschließlich alphanumerische Zeichen (A-Z 0-9 & : = , ; < > . - _) verwenden; dies stellt sicher, dass die Zeichen von allen EBCDIC-Codepages identisch dargestellt werden.

Verwenden Sie nur dann Containernamen, die mit 'DFH' beginnen, wenn Sie hierzu von CICS aufgefordert werden.

Die Option TOCONTAINER kann nur im ersten Befehl **WEB RECEIVE** angegeben werden.

Bei Angabe der Option TOCONTAINER bewahrt CICS den empfangenen Hauptteil der HTTP-Anforderung im 64-Bit-Speicher auf, bis die Anwendung den HTTP-Hauptteil mit einem API-Befehl CONTAINER abrufen kann.

TYPE(cvda)

Gibt den Typ der empfangenen Anforderung an. Die CVDA-Werte lauten wie folgt:

HTTPYES

Gibt an, dass es sich um eine HTTP-Anforderung handelt.

HTTPNO

Gibt an, dass es sich nicht um eine HTTP-Anforderung handelt.

HTTP-Anforderungen und Nicht-HTTP-Anforderungen verwenden unterschiedliche Protokolle, die in TCPIP-SERVICE-Definitionen angegeben sind und daher verschiedene Ports verwenden müssen. Nicht-HTTP-Anforderungen verwenden das benutzerdefinierte Protokoll (USER). Mit der Option TYPE können Sie zwischen den Anforderungstypen unterscheiden, falls Sie dasselbe vom Benutzer geschriebene Anwendungsprogramm für die Beantwortung von HTTP-Anforderungen und Nicht-HTTP-Anforderungen nutzen.

Bedingungen

122 CHANNELERR

Die RESP2-Werte lauten wie folgt:

1

Der in der Option TOCHANNEL angegebene Name enthält ein ungültiges Zeichen oder eine ungültige Zeichenkombination.

110 CONTAINERERR

Die RESP2-Werte lauten wie folgt:

1

Der in der Option TOCONTAINER angegebene Name enthält ein ungültiges Zeichen oder eine ungültige Zeichenkombination.

16 INVREQ

Die RESP2-Werte lauten wie folgt:

1

Der Befehl wurde in einer Anwendung ausgegeben, die nicht zur CICS-Webunterstützung gehört.

14

Die Codepagekombination ist ungültig.

46

Der Wert für die Option SERVERCONV ist ungültig.

80

Die Option CHARACTERSET kann nicht zusammen mit NOSRVCONVERT angegeben werden.

81

Die Option HOSTCODEPAGE kann nicht zusammen mit NOSRVCONVERT angegeben werden.

84

Der Hauptteil ist unvollständig.

145

Der Kanal wurde nicht angegeben und es gibt keinen aktuellen Kanal.

146

Der benannte Container ist schreibgeschützt.

147

Es trat ein interner Konvertierungsfehler auf.

148

Das Benutzerprotokoll wird für Container nicht unterstützt.

149

Die Option TOCONTAINER kann nur im ersten Befehl **WEB RECEIVE** angegeben werden.

22 LENGERR

Die RESP2-Werte lauten wie folgt:

16

Der Wert der Option MAXLENGTH ist kleiner-gleich Null.

36

Es wurde ein partieller Antworthauptteil zurückgegeben. Verwenden Sie weitere Befehle RECEIVE, um den Rest abzurufen.

57

Der Antworthauptteil überschreitet die angegebene Länge und der Rest des Hauptteils wurde gelöscht.

13 NOTFND

Die RESP2-Werte lauten wie folgt:

7

Die Codepage wurde nicht gefunden.

82

Die Client-Codepage (Zeichensatz) wurde nicht gefunden.

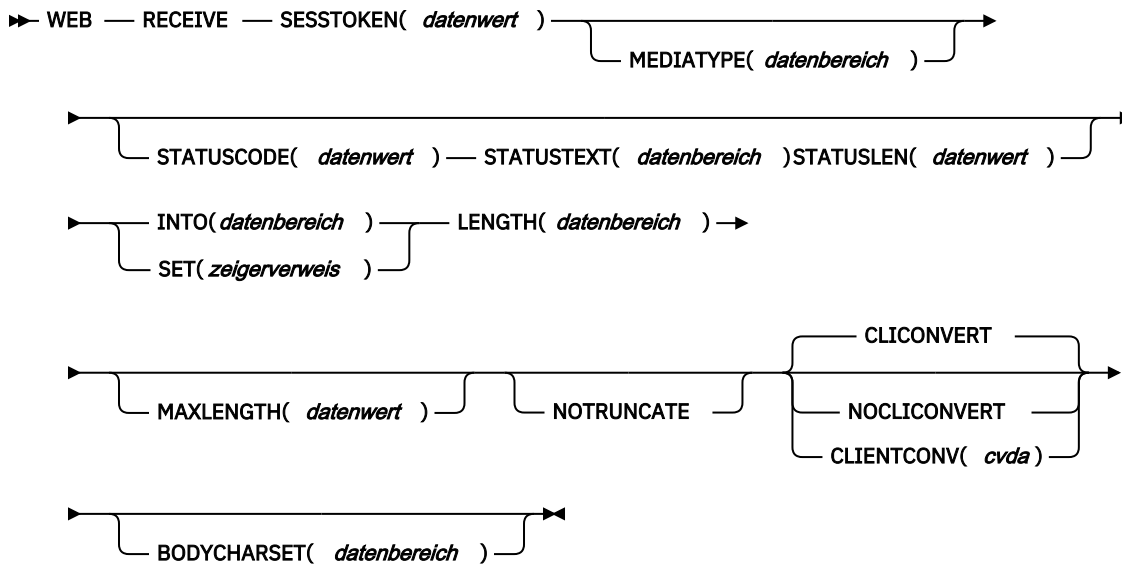
83

Die Host-Codepage (für den Server) wurde nicht gefunden.

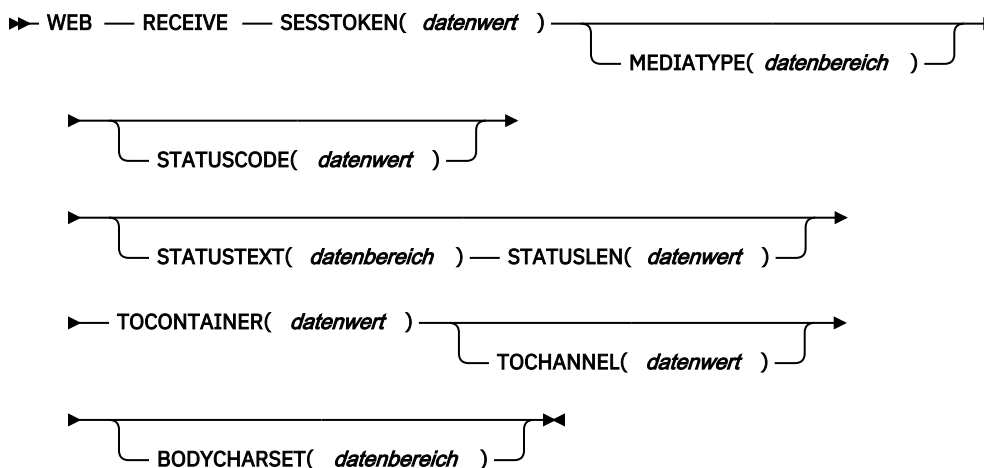
WEB RECEIVE (Client)

Empfängt eine HTTP-Antwort für CICS als HTTP-Client.

WEB RECEIVE (CICS als HTTP-Client mit Pufferverwendung)



WEB RECEIVE (CICS als HTTP-Client mit Containerverwendung)



Bedingungen: CHANNELERR, CONTAINERERR, INVREQ, LENGERR, NOTOPEN, IOERR, TIMEOUT

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl **WEB RECEIVE** für CICS als HTTP-Client empfängt den Hauptteil einer HTTP-Antwort, die von einem Server ausgegeben wurde. Die Header für die HTTP-Antwort können mit dem Befehl **WEB READ HTTPHEADER** oder den Suchbefehlen für HTTP-Header separat geprüft werden. In diesen Befehl muss ein Sitzungstoken einbezogen werden. Anleitungen für die korrekte Verwendung des Befehls **WEB RECEIVE** für CICS als HTTP-Client enthält der Abschnitt [HTTP-Anforderungen über CICS als HTTP-Client absetzen](#).

Wenn ein HTTP-Hauptteil in einem Anwendungspuffer empfangen wird (mithilfe der Optionen **INTO** oder **SET**), besteht beim Befehl **WEB RECEIVE** die Möglichkeit, den Typ der Codepagekonvertierung für eingehende Daten anzugeben, die durch das CICS-Anwendungsprogramm empfangen werden. Falls Sie keine der Optionen für die Codepagekonvertierung angeben (**SERVERCONV**, **CLNTCODEPAGE**, **CHARACTERSET**, **HOSTCODEPAGE**), findet keine Codepagekonvertierung statt.

Beim Empfang eines HTTP-Hauptteils in einem benannten Container ist die Codepagekonvertierung nicht zulässig. Falls der benannte Container durch die Benutzeranwendung vor Ausgabe des Befehls **WEB RECEIVE** erstellt wurde, wird der Container gelöscht und erneut erstellt. Die Informationen zum Medientyp

im Header 'Content-Type' der HTTP-Anforderung bestimmen, ob der Container mit dem Datentyp BIT oder CHAR erneut erstellt wird (es sei denn, die Option CHARACTERSET war angegeben; in diesem Fall geht CICS davon aus, dass die im Container gespeicherten Daten in der Codepage codiert sind, die in der Option CHARACTERSET angegeben ist).

- Falls der Medientyp ein Textmedientyp ist, wird ein Container mit dem Datentyp CHAR erstellt.
- Ist der Medientyp ein nicht für Text bestimmter Medientyp, wird ein Container mit dem Datentyp BIT erstellt.
- Falls die HTTP-Anforderung keine Informationen zum Medientyp enthält, wird standardmäßig vom Textmedientyp ausgegangen.

Falls ein Container mit dem Datentyp CHAR erstellt wird, wird der bei IANA registrierte Name für die ID des codierten Zeichensatzes (Coded Character Set Identifier, CCSID) für die aktuelle Codepage der Daten aus dem Element 'charset' im Header 'Content-Type' der HTTP-Anforderung abgerufen. Falls diese Informationen nicht bereitgestellt oder nicht von CICS unterstützt werden, wird als Standardwert ISO-8859-1 angenommen.

Die Optionen LENGTH, MAXLENGTH, NOTRUNCATE und CLIENTCONV sind nicht zulässig, wenn ein HTTP-Hauptteil in einem benannten Container empfangen wird.

Anmerkung: Der Wert von RTIMOUT, der für die Transaktion angegeben ist, von der die Benutzeranwendung gestartet wird, gibt die Zeit an, die die Anwendung für das Warten auf den Empfang der eingehenden Nachricht vorbereitet ist. (Der Wert von RTIMOUT ist in der Definition des Transaktionsprofils angegeben.) Sobald der von RTIMOUT angegebene Zeitraum abgelaufen ist, gibt CICS eine Antwort TIMEDOUT an die Anwendung zurück. Der Wert Null für RTIMOUT bedeutet, dass die Anwendung unbegrenzt warten kann. Die Standardeinstellung für RTIMOUT bei Definitionen von Transaktionsprofilen ist Null; daher muss diese Einstellung für Anwendungen, die HTTP-Clientanforderungen ausgeben, unbedingt überprüft und geändert werden.

HTTP-Header werden normal unter Verwendung des 64-Bit-Speichers (oberhalb der 2-GB-Grenze) empfangen, verarbeitet und aufbewahrt. Wenn der Hauptteil der HTTP-Clientanforderung in einem Container empfangen wird, werden die Containerdaten unter Verwendung des 64-Bit-Speichers empfangen und verarbeitet. Wenn der HTTP-Hauptteil in einem Anwendungspuffer empfangen wird und CICS diese Daten aufbewahren muss, bevor sie an die Anwendung zurückgegeben werden können, bewahrt CICS die Daten in 64-Bit-Puffern auf.

Optionen

BODYCHARSET(datenbereich)

Gibt den Zeichensatz des HTTP-Antworthauptteils an.

Der Name des Zeichensatzes kann aus bis zu 40 alphanumerischen Zeichen inklusive der entsprechenden Interpunktion bestehen.

Wenn der HTTP-Hauptteil in einem Anwendungspuffer empfangen wird, wird der Zeichensatz wie folgt zurückgegeben:

- Falls die Option INTO oder SET angegeben ist und der HTTP-Hauptteil konvertiert wird, gibt CICS den Zeichensatz des HTTP-Hauptteils vor der Konvertierung zurück.
- Falls die Option INTO oder SET angegeben ist und der HTTP-Hauptteil nicht konvertiert wird, gibt CICS den Zeichensatz zurück, der im Header 'Content-Type' angegeben ist. Wenn keine Zeichensatzinformationen verfügbar sind, werden Leerzeichen zurückgegeben.

Wenn der HTTP-Hauptteil in einem benannten Container empfangen wird, wird der Zeichensatz wie folgt zurückgegeben:

- Falls der Container den Datentyp CHAR besitzt, gibt CICS den Zeichensatz der codierten Daten zurück.
- Falls der Container den Datentyp BIT besitzt, gibt CICS Leerzeichen zurück.

Wenn der zurückgegebene Wert mehr als 40 Byte umfasst, werden die Daten abgeschnitten. Umfasst der zurückgegebene Wert weniger als 40 Byte, werden die Daten rechts mit Leerzeichen aufgefüllt.

CLIENTCONV(*cvda*)

Wenn Sie Daten in einem Puffer empfangen (und entweder die Option INTO oder die Option SET angegeben ist), gibt diese Option an, ob CICS den Entitätshauptteil der Antwort aus dem vom Server verwendeten Zeichensatz in eine für die Anwendung geeignete Codepage konvertiert oder nicht. Standardmäßig wird der Entitätshauptteil konvertiert.

CLICONVERT

CICS konvertiert den Entitätshauptteil der Antwort aus dem vom Server verwendeten Zeichensatz in die Codepage, die Sie für die Anwendung angeben.

NOCLICONVERT

CICS konvertiert nicht den Entitätshauptteil der Antwort, der an die Anwendung in dem vom Server verwendeten Zeichensatz übergeben wird.

Die Angabe eines Zeichensatzes oder einer Anwendungscodepage ist im Befehl WEB RECEIVE nicht zwingend erforderlich, wenn Sie CICS als HTTP-Client einsetzen. Falls eine Codepagekonvertierung erforderlich ist, ermittelt CICS den vom Server verwendeten Zeichensatz anhand des Headers 'Content-Type' in der Nachricht. Falls diese Informationen nicht im Header bereitgestellt werden oder der benannte Zeichensatz von CICS für die Codepagekonvertierung nicht unterstützt wird, wird der Zeichensatz ISO-8859-1 verwendet. Für die Anwendungscodepage wird die Standardcodepage der lokalen CICS-Region (gemäß der Angabe im Systeminitialisierungsparameter LOCALCCSID) oder eine alternative EBCDIC-Codepage verwendet, die Sie im Befehl WEB OPEN angegeben haben.

Für die Ausführung der Codepagekonvertierung bei Verwendung von Puffern (mit Angabe entweder der Option INTO oder der Option SET) gelten bestimmte Hinweise. Wenn Sie Daten in einem Puffer empfangen und die Optionen CHARACTERSET und CLICONVERT nicht angegeben sind, muss der Medientyp für die Nachricht als Dateninhaltstyp Text angeben (gemäß den IANA-Definitionen), damit eine Codepagekonvertierung stattfindet. Für Nachrichten, bei denen kein Medientyp, jedoch die Option CLICONVERT angegeben ist, findet die Codepagekonvertierung ebenfalls statt. Falls ein Medientyp vorhanden ist, bei dem es sich nicht um Text handelt, konvertiert CICS den Nachrichtenhauptteil nicht.

Geben Sie die Option CLIENTCONV nicht an, wenn Sie die Option TOCONTAINER verwenden.

INTO(*datenbereich*)

Gibt den Puffer an, der die empfangenen Daten enthalten soll. Wenn der Parameter INTO angegeben ist, muss auch der Parameter MAXLENGTH mit einem größeren Wert als Null angegeben sein (andernfalls wird an die Anwendung ein Fehler INVREQ mit dem RESP2-Wert 16 zurückgegeben).

Falls die HTTP-Anforderung einen Header 'Content-Length' enthält oder in Blöcken gesendet wurde, bewahrt CICS den HTTP-Hauptteil im internen 64-Bit-Speicher auf; andernfalls bewahrt CICS den Hauptteil im internen 31-Bit-Speicher auf. Der Hauptteil wird aus diesem internen Speicher in den Anwendungspuffer kopiert, der in der Option **INTO** angegeben ist.

LENGTH(*datenbereich*)

Gibt eine Vollwort-Binärzahlvariable an, die auf die Menge der Daten gesetzt wird, die CICS an die Anwendung zurückgegeben hat. Dieser Wert kann - insbesondere bei Einbeziehung eines Doppelbyte- oder Mehrbytezeichensatzes - etwas niedriger als der mit der Option MAXLENGTH festgelegte Grenzwert sein, weil CICS am Ende der Daten kein abgeschnittenes Zeichen zurückgibt.

- Falls die Option NOTRUNCATE **nicht angegeben ist**, sind alle weiteren Daten in der Nachricht jetzt gelöscht worden. Falls weitere Daten vorhanden sind, wird eine Antwort LENGERR mit dem RESP2-Wert 57 zurückgegeben.
- Falls die Option NOTRUNCATE **angegeben ist**, werden alle zusätzlichen Daten aufbewahrt. Falls zusätzliche Daten vorhanden sind, wird eine Antwort LENGERR mit dem RESP2-Wert 36 zurückgegeben. Die Beschreibung für die Option NOTRUNCATE enthält Anweisungen für einen solchen Fall.

Falls Sie zum Speichern des HTTP-Hauptteils einen Anwendungspuffer verwenden, muss die Option LENGTH angegeben sein, wenn die Option INTO oder SET verwendet wird. Geben Sie die Option LENGTH bei Verwendung eines benannten Containers zum Speichern des HTTP-Hauptteils (und somit bei Angabe der Option TOCONTAINER) nicht an.

MAXLENGTH(datenwert)

Gibt als Vollwort-Binärwert die maximale Datenmenge an, die CICS an die Anwendung übergeben soll. Die Option MAXLENGTH gilt unabhängig davon, ob die Option INTO oder SET für den Empfang von Daten angegeben ist. Falls die Daten unter Verwendung von Chunked Transfer Coding gesendet wurden, assembliert CICS die Blöcke in einer einzigen Nachricht, bevor diese an die Anwendung übergeben wird. Die Option MAXLENGTH gilt daher für die Gesamtlänge der in Blöcke aufgeteilten Nachricht und nicht für jeden einzelnen Block. Die Messung der Daten findet nach einer gegebenenfalls erfolgten Codepagekonvertierung statt.

Falls die Länge der Daten den angegebenen Wert überschreitet und die Option NOTRUNCATE **nicht angegeben ist**, werden die Daten auf diesen Wert abgeschnitten und die übrigen Daten werden gelöscht.

Falls die Länge der Daten den angegebenen Wert überschreitet und die Option NOTRUNCATE **angegeben ist**, bewahrt CICS die verbleibenden Daten auf und verwendet sie zur Erfüllung nachfolgender Befehle RECEIVE.

Geben Sie die Option MAXLENGTH nicht an, falls Sie die Option TOCONTAINER verwenden.

MEDIATYPE(datenbereich)

Gibt den Dateninhalt eines bereitgestellten Nachrichtenhauptteils an, z. B. text/xml. Als Wert für die Option MEDIATYPE müssen Sie einen 56 Byte umfassenden Bereich angeben. Der Medientyp kann bis zu 56 alphanumerische Zeichen lang sein, inklusive der entsprechenden Interpunktion. Weitere Informationen zu Medientypen finden Sie unter [IANA-Datenträgertypen und -Zeichensätze](#).

NOTRUNCATE

Gibt an, dass bei einer Überschreitung der in der Option MAXLENGTH angeforderten Länge durch die verfügbaren Daten die übrigen Daten nicht sofort gelöscht, sondern zum Abruf durch nachfolgende Befehle RECEIVE aufbewahrt werden sollen. (Falls keine weiteren Befehle RECEIVE ausgegeben werden, werden die Daten bei der Beendigung der Transaktion gelöscht.)

Ein einzelner Befehl RECEIVE mit der Option SET und ohne die Option MAXLENGTH empfängt alle verbleibenden Daten ungeachtet ihrer Länge. Alternativ können Sie eine Folge von Befehlen RECEIVE mit der Option NOTRUNCATE verwenden, um die verbleibenden Daten in geeigneten Blöcken zu empfangen. Geben Sie so lang Befehle RECEIVE aus, bis Sie keine Antwort LENGERR mehr empfangen. Der Empfang einer kürzeren Länge, als in der Option MAXLENGTH angegeben ist, ist nicht zwangsläufig ein Hinweis auf das Ende der Daten ist. Diese Situation könnte vorliegen, wenn CICS die Rückgabe eines abgeschnittenen Zeichens am Ende der Daten vermeiden muss.

Geben Sie die Option NOTRUNCATE nicht an, falls Sie die Option TOCONTAINER verwenden. Der gesamte HTTP-Hauptteil wird in dem Container gespeichert, der im ersten Befehl **WEB RECEIVE** angegeben war.

SESTOKEN(datenwert)

Gibt den 8 Byte umfassenden Binärwert für das Sitzungstoken an, das eine Verbindung zwischen CICS und einem Server eindeutig kennzeichnet. Dieser Wert wird durch einen Befehl WEB OPEN für CICS als HTTP-Client zurückgegeben. Die Verwendung des Sitzungstokens ist im Abschnitt [Sitzungstokens](#) erläutert.

SET(zeigerverweis)

Gibt einen Zeigerverweis an, der auf die Adresse der empfangenen Daten gesetzt werden soll. Der Zeigerverweis ist gültig, bis der nächste Befehl RECEIVE mit demselben Sitzungstoken ausgegeben wird, oder bis zum Ende der Task.

Falls die HTTP-Anforderung einen Header 'Content-Length' enthält oder in Blöcken gesendet wurde, bewahrt CICS den HTTP-Hauptteil im internen 64-Bit-Speicher auf; andernfalls bewahrt CICS den Hauptteil im internen 31-Bit-Speicher auf. Der Hauptteil wird aus diesem internen Speicher in den Speicher kopiert, der für die Anwendung zugänglich ist, und der Zeigerverweis wird auf diese Kopie gesetzt.

STATUSCODE(datenwert)

Gibt einen Datenbereich für den Empfang des HTTP-Statuscodes an, der durch den Server gesendet wird. Der Code ist ein Halbwort-Binärwert. Beispiele sind 200 (normal) oder 404 (nicht gefunden). Der

Empfang des Statuscodes ist optional. Unter den folgenden Umständen sollten Sie jedoch den Statuscode in jedem Fall empfangen und überprüfen:

- Sie wollen entweder jetzt oder während einer künftigen Verbindung eine identische Anforderung an den Server ausgeben.
- Sie wollen unter Verwendung dieser Verbindung weitere Anforderungen an den Server ausgeben.
- Ihre Anwendung führt eine weitere Verarbeitung aus, die von den Informationen abhängig ist, die Sie in der Antwort empfangen..

Der Abschnitt [HTTP-Statuscodereferenz für CICS-Webunterstützung](#) enthält grundlegende Anleitungen für entsprechende Aktionen, die eine Anwendung als Reaktion auf die Statuscodes für HTTP/1.1 ausführen muss.

STATUSTEXT(datenbereich)

Gibt einen Datenbereich an, der einen vom Server zurückgegebenen Text für die Beschreibung des Statuscodes empfangen soll. Der Text wird als 'Ursachenphrase' bezeichnet. Beispiele sind 'OK' (mit einem Statuscode 200) oder 'Bad Request' (mit einem Statuscode 400). Die zulässige Länge für den Text wird in der Option STATUSLEN angegeben.

STATUSLEN(datenwert)

Gibt als Vollwort-Binärwert die Länge des Datenbereichs an, der einen vom Server zurückgegebenen Text für die Beschreibung des Statuscodes (Option STATUSTEXT) empfangen soll. Der Text wird als 'Ursachenphrase' bezeichnet. Die meisten für HTTP empfohlenen Ursachenphrasen sind kurz, aber für den Fall, dass der Server den empfohlene Ursachencode durch detailliertere Informationen ersetzt, wird hier für den Datenbereich eine Länge von 256 empfohlen.

TOCHANNEL(datenwert)

Gibt den Namen des Kanals an, zu dem der Container gehört. Der Name des Kanals kann aus bis zu 16 alphanumerischen Zeichen inklusive der entsprechenden Interpunktion bestehen. Zulässige Zeichen: A-Z a-z 0-9 \$ @ # / % & ? ! : | " = , ; < > . - und _ . Führende und eingebettete Leerzeichen sind nicht zulässig. Falls der Name weniger als 16 Zeichen umfasst, wird er mit nachgestellten Leerzeichen aufgefüllt. Falls der Kanal nicht vorhanden ist, wird er erstellt. Dieser neue Kanal bleibt gültig, bis sich die Verbindungsebene ändert. Informationen zum Geltungsbereich von Kanälen finden Sie unter [Geltungsbereich eines Kanals](#).

Wenn Sie Ihre Kanäle zwischen CICS-Regionen verlagern wollen, denken Sie daran, dass Sie ausschließlich alphanumerische Zeichen (A-Z 0-9 & : = , ; < > . - _) verwenden; dies stellt sicher, dass die Zeichen von allen EBCDIC-Codepages identisch dargestellt werden.

Sie können den Kanalnamen DFHTRANSACTION angeben, um einen Transaktionskanal zu verwenden. Ein Transaktionskanal verliert nicht seine Gültigkeit, wenn sich die Verbindungsebene ändert; der Zugriff auf diesen Kanal ist in der Transaktion immer möglich. Weitere Informationen finden Sie unter [Kanäle und Container](#).

Falls die Option TOCHANNEL nicht angegeben ist, geht CICS vom aktuellen Kanal aus.

TOCONTAINER(datenwert)

Gibt den Namen des Containers an, in den die Daten gestellt werden. Der Name des Containers kann aus bis zu 16 alphanumerischen Zeichen inklusive der entsprechenden Interpunktion bestehen. Zulässige Zeichen: A-Z a-z 0-9 \$ @ # / % & ? ! : | " = , ; < > . - und _ . Führende und eingebettete Leerzeichen sind nicht zulässig. Falls der Name weniger als 16 Zeichen umfasst, wird er mit nachgestellten Leerzeichen aufgefüllt.

Wenn Sie Ihre Container zwischen CICS-Regionen verlagern wollen, denken Sie daran, dass Sie ausschließlich alphanumerische Zeichen (A-Z 0-9 & : = , ; < > . - _) verwenden; dies stellt sicher, dass die Zeichen von allen EBCDIC-Codepages identisch dargestellt werden.

Verwenden Sie nur dann Containernamen, die mit 'DFH' beginnen, wenn Sie hierzu von CICS aufgefordert werden.

Die Option TOCONTAINER kann nur im ersten Befehl **WEB RECEIVE** angegeben werden.

Bei Angabe der Option TOCONTAINER bewahrt CICS den empfangenen Hauptteil der HTTP-Anforderung im 64-Bit-Speicher auf, bis die Anwendung den HTTP-Hauptteil mit einem API-Befehl CONTAINER abrufen kann.

Bedingungen

122 CHANNELERR

Die RESP2-Werte lauten wie folgt:

1

Der in der Option TOCHANNEL angegebene Name enthält ein ungültiges Zeichen oder eine ungültige Zeichenkombination.

110 CONTAINERERR

Die RESP2-Werte lauten wie folgt:

1

Der in der Option TOCONTAINER angegebene Name enthält ein ungültiges Zeichen oder eine ungültige Zeichenkombination.

19 NOTOPEN

Die RESP2-Werte lauten wie folgt:

27

Das Sitzungstoken ist ungültig.

16 INVREQ

Die RESP2-Werte lauten wie folgt:

10

Der Antwortheader ist ungültig.

15

Bei der Codepagekonvertierung trat ein Fehler auf.

16

Aufgrund einer der folgenden Situationen trat ein Fehler für INTO und MAXLENGTH auf:

- Beide Parameter INTO und MAXLENGTH sind nicht angegeben.
- Beide Parameter INTO und MAXLENGTH sind angegeben, aber der Wert von MAXLENGTH ist kleiner-gleich Null.

22

Es wurde ein ungültiger Block empfangen.

41

Die Verbindung wurde geschlossen. Der Server hat möglicherweise aufgrund von Inaktivität über diese Verbindung ein zulässiges Zeitlimit überschritten.

46

Die Option CLIENTCONV ist ungültig.

67

Der Inhalt der Antwort entspricht nicht dem HTTP-Format. Der Fehler wird generiert, weil ein Syntaxproblem vorliegt.

68

Die mit Chunked Transfer Coding gesendete Nachricht wird gegenwärtig verarbeitet.

71

Beim Chunked Transfer Coding trat ein Fehler auf.

144

Einer oder mehrere Parameter des Webbefehls sind ungültig.

145

Der Kanal wurde nicht angegeben und es gibt keinen aktuellen Kanal.

146

Der benannte Container ist schreibgeschützt.

147

Es trat ein interner Konvertierungsfehler auf.

149

Die Option TOCONTAINER kann nur im ersten Befehl **WEB RECEIVE** angegeben werden.

157

Die HTTP-Header in der Antwort sind länger als 4k; daher kann CICS diese HTTP-Antwort nicht verarbeiten. Die Sitzung ist nicht verwendbar und ein Befehl **WEB CLOSE** ist erforderlich.

22 LENGERR

Die RESP2-Werte lauten wie folgt:

16

Der Wert für MAXLENGTH ist ungültig.

36

Es wurde ein partieller Antworthauptteil zurückgegeben. Verwenden Sie weitere Befehle **RECEIVE**, um den Rest abzurufen.

57

Der Antworthauptteil überschreitet die angegebene Länge und der Rest des Hauptteils wurde gelöscht.

58

Der Statustext überschreitet die angegebene Länge und wurde abgeschnitten.

59

Der Wert für die Option STATUSLEN war nicht größer als Null.

17 IOERR

Die RESP2-Werte lauten wie folgt:

42

Es trat ein Socketfehler auf.

124 TIMEOUT

Die RESP2-Werte lauten wie folgt:

62

Beim Empfangen des Sockets trat eine Zeitlimitüberschreitung auf.

WEB RETRIEVE

Ruft den Wert von DOCTOKEN für ein CICS-Dokument ab, das mit einem Befehl **WEB SEND** gesendet wurde.

WEB RETRIEVE

➡ **WEB — RETRIEVE — DOCTOKEN(*datenbereich*)** ➡

Bedingungen: INVREQ, NOTFND

Dieser Befehl ist threadsicher.

Mit dem Befehl **WEB RETRIEVE** kann eine Anwendung das binäre Token für ein gesendetes Dokument abrufen. Damit das Dokumenttoken durch den Befehl **WEB RETRIEVE** abgerufen werden kann, muss im vorherigen Befehl **WEB SEND** die Option **ACTION(EVENTUAL)** angegeben gewesen sein, damit der Befehl **SEND** beim Abschluss der Anwendung ansteht. Dies liegt daran, dass die Webdomäne bei Verwendung der Option **ACTION(EVENTUAL)** eine Kopie der Informationen für ein Dokument aufbewahrt, nachdem das Dokument gesendet wurde. Bei dem abgerufenen Dokument handelt es sich übrigens um das Doku-

ment, das gesendet wurde; es enthält keine Änderungen, die möglicherweise bis zu dem Zeitpunkt vorgenommen wurden, an dem der Befehl RETRIEVE ausgegeben wird.

Eine gültige Abfolge von Ereignissen für die Ausgabe des Befehls WEB RETRIEVE lautet wie folgt:

```
EXEC CICS WEB SEND  
        ACTION(EVENTUAL)  
EXEC CICS WEB SEND  
        ACTION(EVENTUAL)  
EXEC CICS WEB RETRIEVE  
        DOCTOKEN(MYDOC)
```

Der Wert von DOCTOKEN für den zweiten Befehl WEB SEND wird erfolgreich abgerufen.

Falls der Befehl WEB SEND die Option DOCSTATUS(DOCDELETE) angegeben hat, kann der Befehl WEB RETRIEVE das Dokument nicht abrufen und es wird eine Antwort NOTFND mit dem RESP2-Wert 1 zurückgegeben.

Falls der Befehl WEB SEND die Option ACTION(IMMEDIATE) angegeben hat, kann der Befehl WEB RETRIEVE das Dokument nicht abrufen und es wird eine Antwort NOTFND mit dem RESP2-Wert 1 zurückgegeben. Die Option ACTION(EVENTUAL) wird von der Clientverarbeitung für den Befehl WEB SEND nicht unterstützt, weshalb die Verwendung des Befehls WEB RETRIEVE für Befehle WEB SEND bei einem Client nicht gültig ist.

Optionen

DOCTOKEN(datenbereich)

Gibt einen Puffer an, der das 16 Byte umfassende binäre Token des abzurufenden Dokuments enthält.

Bedingungen

16 INVREQ

RESP2-Werte:

1

Der Befehl wurde in einer Anwendung ausgegeben, die nicht zur CICS-Webunterstützung gehört.

2

Es wurde kein Befehl WEB SEND ausgegeben.

13 NOTFND

RESP2-Werte:

1

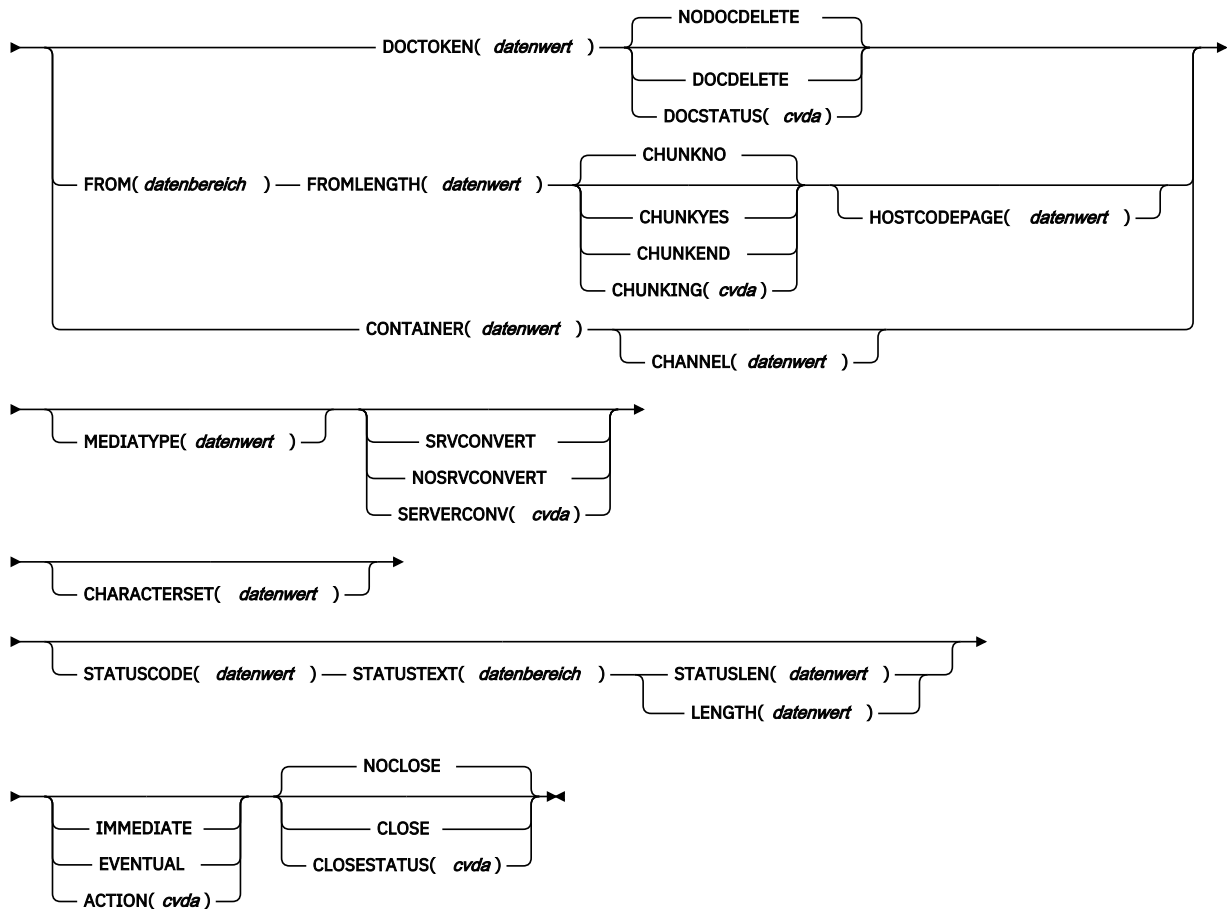
Das Dokument ist nicht verfügbar, weil im letzten Befehl WEB SEND die Option DOCSTATUS(DOCDELETE) angegeben war oder weil der letzte Befehl WEB SEND kein Befehl SEND für ein Dokument mit der Option ACTION(EVENTUAL) war.

WEB SEND (Server)

Sendet eine HTTP-Antwort oder eine Nicht-HTTP-Nachricht.

WEB SEND

► WEB — SEND ►



Bedingungen: CHANNELERR, CONTAINERERR, IOERR, INVREQ, LENGERR, NOTFND

Dieser Befehl ist threadsicher.

Beschreibung

Wenn die CICS-Anwendung die Rolle des Servers innehat, gibt der Befehl **WEB SEND** eine Antwort an, die mithilfe der CICS-Webunterstützung oder der CICS-Geschäftslogikchnittstelle gesendet werden soll. Nachfolgend sind einige Beispiele für mögliche Antworten aufgeführt:

- Antwort auf eine HTTP-Anforderung, die durch einen Web-Client an CICS als HTTP-Server ausgegeben wurde. Anleitungen für die korrekte Verwendung des Befehls WEB SEND für diesen Zweck enthält der Abschnitt [HTTP-Antwort von CICS als HTTP-Server senden](#).
- Von den CICS-Webunterstützungsfunktionen verarbeitete Nicht-HTTP-Nachricht, bei der in der TCP/IP-SERVICE-Definition das benutzerdefinierte Protokoll (USER) angegeben ist. Anweisungen für Nicht-HTTP-Nachrichten finden Sie unter [CICS-Webunterstützung und Nicht-HTTP-Anforderungen](#).
- Antwort auf eine Anforderung aus einer anderen Anwendung, die unter Verwendung der CICS-Geschäftslogikchnittstelle direkt Kontakt zum Programm aufgenommen hat, statt hierzu den HTTP-Listener von CICS einzusetzen. Anweisungen für die CICS-Geschäftslogikchnittstelle enthält der Abschnitt [Einführung in die CICS-Geschäftslogikchnittstelle](#).

Während einer Task kann nur eine einzige Antwort gesendet werden. Hierbei kann es sich um eine Standardantwort unter Verwendung eines einzigen Befehls WEB SEND oder um eine in Blöcke aufgeteilte Antwort handeln, für die eine Folge von Befehlen WEB SEND verwendet wird.

Falls Sie versuchen, während derselben Task eine zweite Antwort zu senden, ist das Ergebnis davon abhängig, ob die Option IMMEDIATE oder die Option EVENTUAL im Befehl WEB SEND für die erste Antwort angegeben war.

- Falls die Option IMMEDIATE für die erste Antwort verwendet wurde, wird ein Fehler zurückgegeben, wenn Sie versuchen, die zweite Antwort zu senden.
- Falls die Option EVENTUAL für die erste Antwort verwendet wurde, überschreibt die zweite Antwort den Inhalt der vorherigen Antwort (Statuszeile, HTTP-Header und Nachrichtenhauptteil). Die erste Antwort geht verloren und die zweite Antwort wird gesendet.

Bei jedem Empfang einer Antwort von einem Web-Client startet CICS eine neue Task für die Verarbeitung der Antwort.

Falls der Hauptteil der HTTP-Serveranforderung in einem Container aufbewahrt wird, bevor er an den Server gesendet wird, bewahrt CICS den Hauptteil im 64-Bit-Speicher auf und sendet ihn von dort aus. Für den Fall, dass der Hauptteil der HTTP-Serveranforderung in einem Anwendungspuffer aufbewahrt wird und vor dem Senden kopiert oder konvertiert werden muss, bewahrt CICS den Hauptteil im 64-Bit-Speicher auf und sendet ihn von dort aus. Beim Senden eines Dokuments wird das Dokument jedoch im 31-Bit-Speicher verarbeitet.

Optionen

ACTION(*cvda*)

Gibt an, wie die Nachricht gesendet werden soll. Beim Einsatz von CICS als HTTP-Server gelten die folgenden CVDA-Werte:

IMMEDIATE

Sendet die Antwort sofort an den Web-Client. Bei Angabe der Option CHUNKING wird von der Option IMMEDIATE ausgegangen.

EVENTUAL

Sendet die Antwort an den Web-Client am Ende der Task. Bei Angabe der Option CHUNKING wird die Option EVENTUAL ignoriert. Für gesendete Nachrichten, bei denen das Chunked Transfer Coding nicht verwendet wird, ist EVENTUAL der Standardwert.

CHANNEL(*datenwert*)

Gibt den Namen des Kanals an, zu dem der Container gehört. Der Name des Kanals kann aus bis zu 16 alphanumerischen Zeichen inklusive der entsprechenden Interpunktion bestehen. Führende und eingebettete Leerzeichen sind nicht zulässig. Falls der Name weniger als 16 Zeichen umfasst, wird er mit nachgestellten Leerzeichen aufgefüllt. Sie können den Kanalnamen DFHTRANSACTION angeben, um den Transaktionskanal zu verwenden.

Falls die Option CONTAINER angegeben ist, ist die Option CHANNEL optional.

Ist die Option CHANNEL nicht angegeben, geht CICS vom aktuellen Kanal aus.

CHARACTERSET(*datenwert*)

Gibt einen Zeichensatz an, in den CICS den Entitätshauptteil des mit dem Befehl gesendeten Elements vor dem Sendevorgang konvertiert. Der Name des Zeichensatzes kann aus bis zu 40 alphanumerischen Zeichen inklusive der entsprechenden Interpunktion bestehen. CICS unterstützt nicht alle von IANA genannten Zeichensätze. Im Abschnitt HTML-codierte Zeichensätze sind die IANA-Zeichensätze aufgelistet, die durch CICS für die Codepagekonvertierung unterstützt werden.

Wenn die Option CHARACTERSET angegeben ist, wird die Option SRVCONVERT angenommen, weshalb eine Codepagekonvertierung für den Entitätshauptteil stattfindet. Statt den Zeichensatz selbst auszuwählen, können Sie auch die Option SRVCONVERT und/oder die Option HOSTCODEPAGE (sofern zulässig) auswählen und die Option CHARACTERSET weglassen, damit CICS einen passenden Zeichensatz für den Nachrichtenhauptteil festlegt. Die Beschreibung für die Option SERVERCONV enthält Anweisungen für einen solchen Fall.

Falls Sie keine der Optionen für die Codepagekonvertierung angeben, findet keine Codepagekonvertierung statt.

CHUNKING(*cvda*)

Wird zum Steuern des Nachrichtensendevorgangs verwendet, wenn die Nachricht in Blöcke aufgeteilt gesendet wird (dies wird als 'Chunked Transfer Coding' bezeichnet). Wenn die Option nicht angegeben ist, wird das Chunked Transfer Coding standardmäßig nicht verwendet. Das Chunked Transfer Coding ist nur bei Clients mit HTTP/1.1 zulässig und kann für Clients mit HTTP/1.0 oder Nicht-HTTP-Nachrichten nicht verwendet werden.

Der Inhalt einer in Blöcke aufgeteilten Nachricht kann auf diejenige Weise in Blöcke aufgeteilt werden, die für das Anwendungsprogramm am besten geeignet ist. Der Hauptteil einer in Blöcke aufgeteilten Nachricht kann nicht direkt aus CICS-Dokumenten gebildet werden, weshalb die Option DOC-TOKEN nicht verwendet werden kann.

Verwenden Sie für jeden Block der Nachricht einen separaten Befehl WEB SEND mit der Option CHUN-KYES. Geben Sie mit der Option FROM den Datenblock und mit der Option FROMLENGTH die Länge des Blocks an. Weitere Optionen für die Nachricht (z. B. die Option CLOSESTATUS) können im ersten Befehl WEB SEND der Befehlsfolge (mit dem der erste Block gesendet wird) angegeben werden; geben Sie jedoch in nachfolgenden Befehlen (mit denen der zweite und darauffolgende Blöcke gesendet werden) keine Optionen an.

Sobald Sie den letzten Datenblock gesendet haben, geben Sie einen weiteren Befehl WEB SEND mit der Option CHUNKEND und ohne die Option FROM oder FROMLENGTH an. CICS sendet dann einen leeren Block an den Empfänger, um die in Blöcke aufgeteilte Nachricht abzuschließen.

Falls einer der Befehle WEB SEND während der Befehlsfolge fehlschlägt, wird eine Fehlerantwort zurückgegeben und nachfolgende Sendevorgänge schlagen ebenfalls fehl. Diese Situation sollte von der Anwendung auf geeignete Weise abgewickelt werden. Falls alle Blöcke erfolgreich gesendet wurden, die Anwendung jedoch nicht den abschließenden Befehl WEB SEND mit der Option CHUNKEND ausgibt, wird die Transaktion mit dem Abbruchcode AWBP abgebrochen. Eine unvollständige in Blöcke aufgeteilte Nachricht sollte vom Empfänger ignoriert und gelöscht werden.

Der Abschnitt HTTP-Anforderung oder -Antwort mit Chunked Transfer Coding senden enthält eine umfassende Beschreibung der Prozedur für das Chunked Transfer Coding, die Sie befolgen sollten, damit Ihre in Blöcke aufgeteilte Nachricht vom Empfänger angenommen werden kann.

Die CVDA-Werte lauten wie folgt:

CHUNKNO

Das Chunked Transfer Coding wird nicht für die Nachricht verwendet. Dies ist der Standardwert, wenn die Option CHUNKING nicht angegeben ist.

CHUNKYES

Das Chunked Transfer Coding wird verarbeitet. Die durch die Option FROM angegebenen Daten stellen einen Block der Nachricht dar.

CHUNKEND

Das Chunked Transfer Coding ist abgeschlossen. Für diesen Sendevorgang werden keine Daten angegeben. CICS sendet einen leeren Block an den Empfänger, um die in Blöcke aufgeteilte Nachricht abzuschließen.

Geben Sie bei Verwendung der Option CONTAINER die Option CHUNKING nicht an. Eine in Blöcke aufgeteilte Antwort kann nicht aus einem Container gesendet werden.

CLNTCODEPAGE(*datenwert*)

Diese Option wird nur zu Upgradezwecken unterstützt. Sie wurde durch die Option CHARACTERSET ersetzt. Bei beiden Schlüsselwörtern führt CICS dieselbe Aktion aus.

CLOSESTATUS(*cvda*)

Gibt an, ob CICS die Verbindung nach dem Senden der Nachricht schließt oder nicht. Standardmäßig wird die Verbindung nicht geschlossen. Die CVDA-Werte lauten wie folgt:

CLOSE

CICS schreibt für diese Antwort einen Verbindungsheader mit der Verbindungsoption 'close' (Connection: close) und schließt die Verbindung zum Web-Client, nachdem die Antwort gesendet wurde. Der Header benachrichtigt den Web-Client über den Abschluss. (Bei einem Client mit Version

HTTP/1.0 erzielt CICS dieselbe Wirkung, indem der Header 'Connection: Keep-Alive' weggelassen wird.)

Falls das Chunked Transfer Coding verwendet wird, kann die Option CLOSE für den ersten Block der Nachricht angegeben werden, um den Web-Client darüber zu informieren, dass die Verbindung nach Abschluss der in Blöcke aufgeteilten Nachricht geschlossen wird.

NOCLOSE

Bedeutet, dass der Header 'Connection: close' für diese Antwort nicht verwendet wird und die Verbindung geöffnet bleibt. Falls für den Web-Client die Version HTTP/1.0 erkannt wird und der Client einen Verbindungsheader mit der Verbindungsoption 'Keep-Alive' (Connection: Keep-Alive) gesendet hat, sendet CICS denselben Header, um kenntlich zu machen, dass eine persistente Verbindung erhalten bleibt.

CONTAINER(datenwert)

Gibt den Namen des Containers an, in dem der HTTP-Hauptteil aufbewahrt wird, bevor er an den Server gesendet wird. Der Name des Containers kann aus bis zu 16 alphanumerischen Zeichen inklusive der entsprechenden Interpunktion bestehen. Führende und eingebettete Leerzeichen sind nicht zulässig. Falls der Name weniger als 16 Zeichen umfasst, wird er mit nachgestellten Leerzeichen aufgefüllt.

Bei Angabe der Option CONTAINER speichert CICS den HTTP-Hauptteil im 64-Bit-Speicher und sendet ihn von dort aus.

DOCSTATUS(cvda)

Gibt an, ob das Dokument während der Verarbeitung des Befehls WEB SEND gelöscht werden soll oder nicht. Die CVDA-Werte lauten wie folgt:

DOCDELETE

CICS löscht das Dokument, nachdem der Dokumentinhalt für den Sendevorgang gespeichert wurde. Der dem Dokument zugeordnete Speicher wird sofort freigegeben. Falls Sie nachfolgende Anforderungen für das Dokument ausgeben, generieren diese eine Antwort NOTFND.

NODOCDELETE

CICS löscht das Dokument während der Verarbeitung des Befehls WEB SEND nicht. Dies ist der Standardwert für die Option DOCSTATUS.

DOCTOKEN(datenwert)

Gibt das 16 Byte umfassende binäre Token eines Dokuments an, das als Nachrichtenhauptteil gesendet werden soll. Das Dokument wird mit der CICS-Dokumentschnittstelle unter Verwendung der EXEC CICS-Befehle **DOCUMENT CREATE**, **INSERT** und **SET** erstellt. Als alternatives Verfahren für die Erstellung eines Nachrichtenhauptteils steht die Option FROM zur Verfügung.

Der Hauptteil einer in Blöcke aufgeteilten Nachricht kann nicht aus CICS-Dokumenten gebildet werden, weshalb die Option DOCTOKEN für das Chunked Transfer Coding nicht verwendet werden kann.

Bei Angabe der Option DOCTOKEN wird das Dokument im 31-Bit-Speicher verarbeitet.

FROM(datenbereich)

Gibt einen Puffer von Daten an, in dem sich der vollständige Nachrichtenhauptteil oder ein Block des Nachrichtenhauptteils befindet. Der Nachrichtenhauptteil wird durch das Anwendungsprogramm erstellt. Wenn Sie die Option FROM angeben, müssen Sie mit der Option FROMLENGTH die Länge des Datenpuffers angeben. Als alternatives Verfahren für die Erstellung eines Nachrichtenhauptteils steht die Option DOCTOKEN zur Verfügung; diese Option kann jedoch nicht für den Hauptteil einer in Blöcke aufgeteilten Nachricht verwendet werden.

Wenn die Option FROM angegeben ist und CICS den HTTP-Hauptteil vor dem Senden kopieren oder konvertieren muss, wird der Hauptteil im 64-Bit-Speicher kopiert oder konvertiert und von dort aus gesendet.

Für die Größe des Datenbereichs gibt es keinen Maximalwert; aufgrund von Speicheraspekten ist seine Größe in der Praxis jedoch begrenzt. Der Abschnitt Entitätshauptteil für eine HTTP-Nachricht erstellen enthält hierzu weitere Informationen.

FROMLENGTH(datenwert)

Gibt als Vollwort-Binärwert die Länge des Datenpuffers an, der in der Option FROM angegeben ist. Dieser Wert muss unbedingt korrekt angegeben werden, weil eine falsche Datenlänge für den Empfänger der Nachricht zu Problemen führen kann.

HOSTCODEPAGE(datenwert)

Gibt den 8 Zeichen langen Namen der CICS-Codepage (für den Host) an, die vom Anwendungsprogramm für den Entitätshauptteil verwendet wurde.

Das CICS-Standardformat für einen Codepagenamen besteht aus der Codepagenummer (oder allgemeiner der CCSID), die nach Bedarf mithilfe von 3 bis 5 Dezimalziffern geschrieben und anschließend mit nachgestellten Leerzeichen auf 8 Zeichen aufgefüllt wird. Bei der Codepage 37, die aus weniger als 3 Ziffern besteht, lautet das Standardformat '037'. CICS akzeptiert jetzt als Codepagenamen auch eine beliebige Dezimalzahl mit bis zu 8 Ziffern (aufgefüllt mit nachgestellten Leerzeichen) im Bereich von 1 bis 65535, auch wenn dieser nicht im Standardformat angegeben ist.

Wenn die Option HOSTCODEPAGE angegeben ist, wird die Option SRVCONVERT angenommen, weshalb eine Codepagekonvertierung für den Entitätshauptteil stattfindet. Wenn Sie die Option SRVCONVERT und/oder CHARACTERSET angeben, die Option HOSTCODEPAGE jedoch nicht, wird CICS die Ermittlung der Host-Codepage überlassen.

Falls zur Bildung des Antworthauptteils ein CICS-Dokument verwendet wird (Option DOCTOKEN), geben Sie die Option HOSTCODEPAGE nicht an, weil CICS die Host-Codepage aus dem CICS-Dokumentdomänendatensatz der Host-Codepages für das Dokument ermittelt.

Wird der Antworthauptteil aus einem Puffer mit Daten gebildet (Option FROM), können Sie die Option HOSTCODEPAGE angeben. Wenn diese Option nicht angegeben ist, ist der Standardwert die Standardcodepage für die lokale CICS-Region, die im Systeminitialisierungsparameter LOCALCCSID festgelegt wurde. Wenn Sie eine Codepagekonvertierung benötigen, Ihre Anwendung jedoch eine andere Host-Codepage verwendet hat, geben Sie sie mit der Option HOSTCODEPAGE an.

Falls Sie keine der Optionen für die Codepagekonvertierung angeben, findet keine Codepagekonvertierung statt.

Geben Sie bei Verwendung der Option CONTAINER die Option HOSTCODEPAGE nicht an.

LENGTH(datenwert)

Diese Option wird nur zu Upgradezwecken unterstützt. Sie wurde durch die Option STATUSLEN ersetzt.

MEDIATYPE(datenwert)

Gibt den Dateninhalt eines bereitgestellten Nachrichtenhauptteils an, z. B. text/xml. Als Wert für die Option MEDIATYPE müssen Sie einen 56 Byte umfassenden Bereich angeben. Der Medientyp kann bis zu 56 alphanumerische Zeichen lang sein, inklusive der entsprechenden Interpunktion, jedoch ohne Leerzeichen. Weitere Informationen zu Medientypen finden Sie unter [IANA-Datenträgertypen und -Zeichensätze](#). CICS überprüft, ob das Format des Medientyps richtig ist, nimmt jedoch keine Überprüfung für die Gültigkeit des Medientyps in Bezug auf den Dateninhalt vor. CICS stellt keinen Standardwert bereit. Unter bestimmten Umständen wirkt sich der von Ihnen angegebene Medientyp darauf aus, ob eine Codepagekonvertierung stattfindet oder nicht; weitere Informationen hierzu können Sie der Beschreibung für die Option SERVERCONV entnehmen.

SERVERCONV(cvda)

Gibt an, ob CICS den Entitätshauptteil des mit dem Befehl gesendeten Elements vor dem Sendevorgang aus der von der Anwendung verwendeten Codepage in einen für den Empfänger geeigneten Zeichensatz konvertiert oder nicht. Mit den Optionen CHARACTERSET und HOSTCODEPAGE können Sie in diesem Befehl den Zeichensatz und die Codepage angeben, die verwendet werden. Falls Sie eine dieser Optionen angeben, wird die Codepagekonvertierung (Option SRVCONVERT) angenommen. Alternativ können Sie eine oder beide dieser Optionen weglassen, die Option SRVCONVERT angeben und CICS einen passenden Zeichensatz und eine geeignete Codepage festlegen lassen.

SRVCONVERT

CICS konvertiert den Entitätshauptteil der Nachricht.

Wenn Sie die Option SRVCONVERT ohne die Option CHARACTERSET angeben, ermittelt CICS folgendermaßen einen geeigneten Zeichensatz:

1. Falls die Anforderung des Web-Clients einen Header 'Content-Type' enthält, der einen von CICS unterstützten Zeichensatz angibt, wird dieser Zeichensatz verwendet.
2. Falls die Anforderung des Web-Clients keinen Header 'Content-Type' enthält oder der benannte Zeichensatz nicht unterstützt wird, wird der Zeichensatz ISO-8859-1 verwendet.
3. Bei Nachrichten, die keine HTTP-Nachrichten sind (und mit dem für USER angegebenen Protokoll gesendet wurden), wird der Zeichensatz ISO-8859-1 verwendet.

Wenn Sie die Option SRVCONVERT ohne die Option HOSTCODEPAGE angeben, ermittelt CICS die Host-Codepage folgendermaßen:

- Bei Verwendung der Option FROM ermittelt CICS als Host-Codepage die Standardcodepage für die lokale CICS-Region, die im Systeminitialisierungsparameter LOCALCCSID angegeben wurde.
- Falls die Option DOCTOKEN verwendet wird, ermittelt CICS die Host-Codepage aus dem CICS-Dokumentdomänen Datensatz der Host-Codepages für das Dokument.
- Wenn die Option CONTAINER verwendet wird, ermittelt CICS als Host-Codepage diejenige Codepage, die für die Datencodierung verwendet wurde, als der HTTP-Hauptteil im Container gespeichert wurde.

Falls Sie die Option SRVCONVERT allein angeben, müssen Sie beachten, dass die Option MEDIATYPE einen Dateninhaltstyp angeben muss, der gemäß den IANA-Definitionen als Text erkannt werden kann, damit eine Codepagekonvertierung stattfindet. Für Medientypen, bei denen es sich nicht um Text handelt, konvertiert CICS den Nachrichtenhauptteil nicht. Zur Kompatibilität mit webfähigen Anwendungen, die in früheren Releases codiert wurden, hat die Option MEDIATYPE jedoch keinen Einfluss auf die Codepagekonvertierung, falls Sie eine der Optionen CHARACTERSET oder HOSTCODEPAGE angeben oder die Option SERVERCONV weglassen.

Container mit dem Datentyp BIT enthalten keine Textdaten und unterstützen daher keine Codepagekonvertierung. Infolgedessen wird ein Fehler INVREQ mit einem RESP2-Wert erzeugt, wenn Sie eine der Optionen SRVCONVERT oder CHARACTERSET mit einem Container des Datentyps BIT codieren.

NOSRVCONVERT

CICS konvertiert den Entitätshauptteil der HTTP-Anforderung nicht; dieser wird in dem von der Anwendung verwendeten Zeichensatz an den Server gesendet. Falls Sie die Option NOSRVCONVERT angeben, ist eine Angabe der Optionen CHARACTERSET oder HOSTCODEPAGE nicht möglich.

Anmerkung: Falls Sie keine der Optionen für die Codepagekonvertierung angeben (SERVERCONV, CLNTCODEPAGE, CHARACTERSET, HOSTCODEPAGE), findet keine Codepagekonvertierung statt.

STATUSCODE(datenwert)

Gibt einen HTTP-Standardstatuscode an, der vom Anwendungsprogramm bestimmt wird und in die Statuszeile der HTTP-Antwort eingefügt werden soll. Bei diesem Code handelt es sich um einen Halbwort-Binärwert. Beispiele sind 200 (normale Antwort) oder 404 (nicht gefunden). Falls diese Option nicht angegeben ist, stellt CICS den Standardwert 200 bereit.

Der Abschnitt [HTTP-Statuscodereferenz für CICS-Webunterstützung](#) enthält Informationen zur Verwendung von Statuscodes für die CICS-Webunterstützung. Bei den Statuscodes 204, 205 und 304 ist ein Nachrichtenhauptteil nicht zulässig und CICS gibt eine Fehlerantwort für den Befehl zurück, falls Sie versuchen, einen Nachrichtenhauptteil einzubeziehen. Ansonsten überprüft CICS nicht, ob Ihre Verwendung des Statuscodes angemessen ist.

STATUSLEN(datenwert)

Gibt als Vollwort-Binärwert die Länge der Zeichenfolge an, die in der Option STATUSTEXT angegeben ist.

STATUSTEXT(datenbereich)

Gibt einen Datenbereich an, der einen lesbaren Text enthält, mit dem die Ursache für den Statuscode beschrieben wird. Der Text wird als 'Ursachenphrase' bezeichnet. Beispiele sind 'OK' (mit einem Sta-

tuscode 200) oder 'Bad Request' (mit einem Statuscode 400). In der Spezifikation von HTTP/1.1 (RFC 2616) ist für jeden Statuscode eine empfohlene Ursachenphrase definiert; ihre Verwendung ist jedoch nicht zwingend erforderlich.

Bedingungen

122 CHANNELERR

Die RESP2-Werte lauten wie folgt:

2

Der in der Option CHANNEL angegebene Kanal wurde nicht gefunden.

110 CONTAINERERR

Die RESP2-Werte lauten wie folgt:

2

Der in der Option CONTAINER angegebene Container wurde nicht gefunden.

16 INVREQ

Die RESP2-Werte lauten wie folgt:

1

Der Befehl wurde in einer Anwendung ausgegeben, die nicht zur CICS-Webunterstützung gehört.

11

Der Aktionscode ist ungültig.

13

Der Status für den Schließvorgang ist ungültig.

14

Die Codepagekombination ist ungültig.

32

Der Medientyp ist ungültig.

41

Die Verbindung wurde geschlossen.

46

Der Wert für die Option SERVERCONV ist ungültig.

72

Der Statuscode unterstützt keinen Nachrichtenhauptteil.

75

Die Sendefolge ist ungültig.

77

Der Block ist unvollständig.

80

Die Option CHARACTERSET kann nicht zusammen mit NOSRVCONVERT angegeben werden.

81

Die Option HOSTCODEPAGE kann nicht zusammen mit NOSRVCONVERT angegeben werden.

85

Bei Nicht-HTTP-Nachrichten ist das Chunking (Aufteilung in Blöcke) nicht möglich.

86

Bei Clients mit der Version HTTP/1.0 kann das Chunking nicht verwendet werden.

87

Der Statuscode ist nicht zulässig.

88

Die Host-Codepage ist nicht zulässig.

- 89**
Vorherige Sendevorgänge über diese Verbindung sind fehlgeschlagen. Weitere Sendevorgänge sind nicht zulässig.
- 90**
Die Optionen STATUSCODE und STATUSTEXT sind für den zweiten und alle nachfolgenden Blöcke nicht zulässig.
- 91**
Die Optionen CHARACTERSET und CLNTCODEPAGE sind für den zweiten und alle nachfolgenden Blöcke nicht zulässig.
- 92**
Die Option HOSTCODEPAGE ist für den zweiten und alle nachfolgenden Blöcke nicht zulässig.
- 93**
Die Option MEDIATYPE ist für den zweiten und alle nachfolgenden Blöcke nicht zulässig.
- 94**
Die Option CLOSESTATUS ist für den zweiten und alle nachfolgenden Blöcke nicht zulässig.
- 95**
Die Option SERVERCONV ist für den zweiten und alle nachfolgenden Blöcke nicht zulässig.
- 120**
Der Wert für die Option CHUNKING ist ungültig.
- 121**
Die Option FROMLENGTH ist erforderlich.
- 122**
Die Option FROM ist erforderlich.
- 123**
Es wurde kein Nachrichtenhauptteil angegeben. Verwenden Sie die Option FROM, DOCTOKEN oder CHUNKEND.
- 124**
Die Option CHUNKING war nicht angegeben, die Option FROMLENGTH ist erforderlich.
- 125**
Die Option CHUNKNO war angegeben, die Option FROM ist erforderlich.
- 126**
Die Option CHUNKNO war angegeben, die Option FROMLENGTH ist erforderlich.
- 127**
Die Option CHUNKYES war angegeben, die Option FROM ist erforderlich.
- 128**
Die Option CHUNKYES war angegeben, die Option FROMLENGTH ist erforderlich.
- 129**
Die Option FROM ist bei Verwendung der Option CHUNKEND nicht zulässig.
- 130**
Die Option FROMLENGTH ist bei Verwendung der Option CHUNKEND nicht zulässig.
- 131**
Für die Option FROMLENGTH war Null angegeben.
- 143**
Der angegebene Wert für DOCSTATUS ist ungültig.
- 145**
Der Kanal wurde nicht angegeben und es gibt keinen aktuellen Kanal.
- 147**
Es trat ein interner Konvertierungsfehler auf.
- 148**
Das Benutzerprotokoll wird für Container nicht unterstützt.

150

Die Konvertierung wurde angefordert, aber die zu sendenden Daten befinden sich in einem Container mit dem Datentyp BIT.

151

Das Chunking ist während der Webfehlerbehandlung ungültig.

152

Die Option ACTION(EVENTUAL) ist während der Webfehlerbehandlung ungültig.

17 IOERR

Die RESP2-Werte lauten wie folgt:

42

Es trat ein Socketfehler auf.

22 LENGERR

Die RESP2-Werte lauten wie folgt:

50

Der Wert für die Option FROMLENGTH war nicht größer als Null.

13 NOTFND

Die RESP2-Werte lauten wie folgt:

1

Das Dokument wurde nicht erstellt bzw. wurde gelöscht oder der Name war falsch angegeben.

7

Die Client-Codepage (Zeichensatz) wurde nicht gefunden.

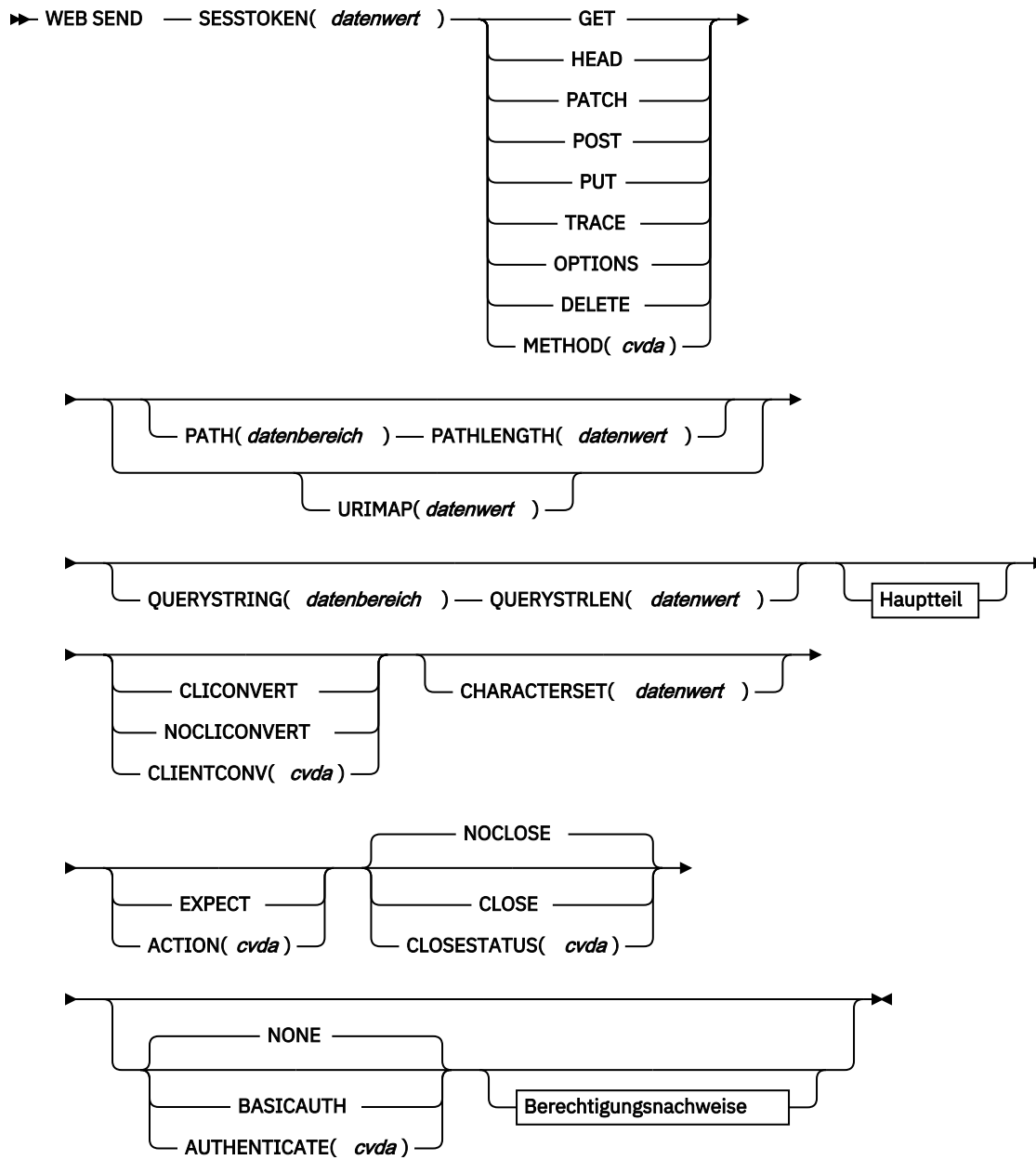
83

Die Host-Codepage (für den Server) wurde nicht gefunden.

WEB SEND (Client)

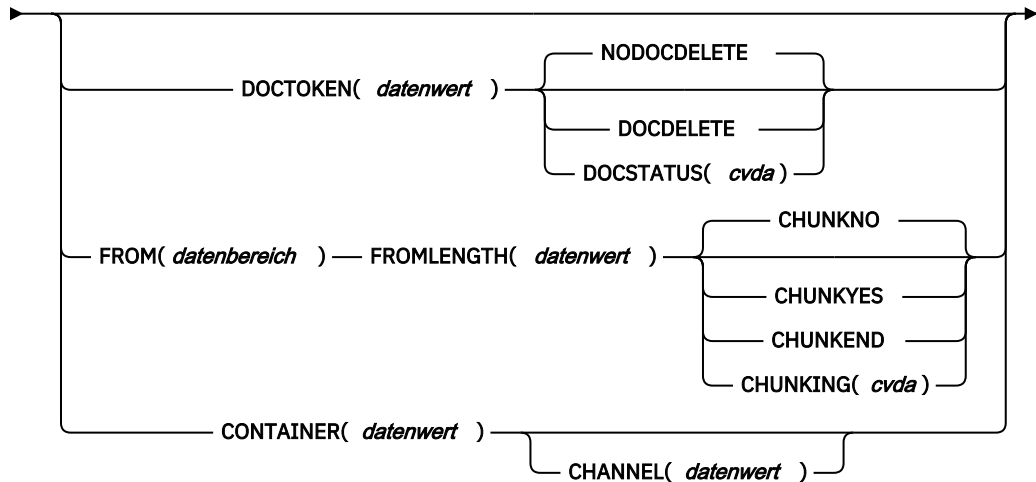
Sendet durch CICS als HTTP-Client eine HTTP-Anforderung unter Verwendung der CICS-Webunterstützung.

WEB SEND (CICS als HTTP-Client)

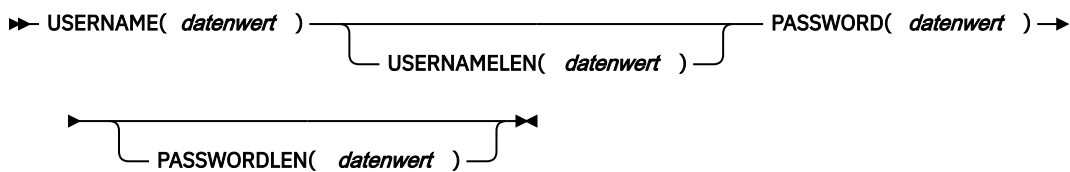


Hauptteil

➔ MEDIATYPE(*datenwert*) ➔



Berechtigungsnachweise



Bedingungen: CHANNELERR, CONTAINERERR, INVREQ, IOERR, LENGERR, NOTAUTH, NOTFND, NOT-OPEN, TIMEOUT, TOKENERR

Dieser Befehl ist threadsicher.

Beschreibung

Mit dem Befehl WEB SEND für CICS als HTTP-Client wird eine HTTP-Anforderung an einen Server ausgegeben. In diesen Befehl muss ein Sitzungstoken einbezogen werden. Anleitungen für die korrekte Verwendung des Befehls WEB SEND für CICS als HTTP-Client enthält der Abschnitt [HTTP-Anforderungen über CICS als HTTP-Client absetzen](#).

Der Befehl WEB SEND kann für CICS als HTTP-Client nicht verwendet werden, nachdem die Verbindung zum Server geschlossen wurde. Diese Situation kann eintreten, wenn entweder das Anwendungsprogramm oder der Server einen Header 'Connection: close' in einer Nachricht sendet. Falls Sie testen müssen, ob der Server das Beenden der Verbindung angefordert hat, suchen Sie mit dem Befehl WEB READ HTTPHEADER in der letzten Nachricht vom Server nach dem Header 'Connection: close'.

Wenn Sie CICS als HTTP-Client nutzen, können Sie den Befehl WEB CONVERSE als Alternative zur Ausgabe eines Befehls WEB SEND mit einem anschließenden Befehl WEB RECEIVE verwenden. Der Befehl WEB CONVERSE unterstützt jedoch das Chunked Transfer Coding nicht, da dieses Verfahren eine Folge von Sendeaktionen erforderlich macht und der Befehl WEB CONVERSE nur eine einzige Sendeaktion bereitstellt.

Die Anforderung kann ein Zeitlimit überschreiten, wenn eine Nachricht an den Server gesendet wird. In diesem Fall gilt das Zeitlimitintervall für Deadlocks, das im Attribut DTIMOUT der Transaktionsdefinition angegeben ist, und CICS gibt eine Antwort TIMEOUT an die Anwendung zurück.

HTTP-Header werden normal unter Verwendung des 64-Bit-Speichers (oberhalb der 2-GB-Grenze) aufbewahrt, verarbeitet und gesendet. Wenn der Hauptteil der HTTP-Clientanforderung aus einem Container gesendet wird, werden die Containerdaten unter Verwendung des 64-Bit-Speichers verarbeitet und gesendet. Wenn der HTTP-Hauptteil aus einem Anwendungspuffer gesendet wird und CICS die Daten konvertieren muss, werden die Daten in einem 64-Bit-Puffer konvertiert und von dort aus gesendet. Beim

Senden eines Dokuments wird das Dokument jedoch im 31-Bit-Speicher (oberhalb der 16-MB-Grenze) verarbeitet.

Optionen

ACTION(*cvda*)

Gibt an, wie die Nachricht gesendet werden soll. Dieser CVDA-Wert gilt beim Einsatz von CICS als HTTP-Client:

EXPECT

Weist CICS an, einen Header 'Expect' zusammen mit der Anforderungszeile und den Headern für die Anforderung zu senden und eine Antwort '100-Continue' abzuwarten, bevor der Nachrichtenhauptteil an den Server gesendet wird. Falls eine andere Antwort als '100-Continue' empfangen wird, informiert CICS das Anwendungsprogramm und bricht den Sendevorgang ab. Falls nach einer Wartezeit keine Antwort empfangen wird, sendet CICS den Nachrichtenhauptteil.

Der Header 'Expect' wird von Servern mit einer älteren Version als HTTP 1.1 nicht unterstützt. Falls CICS die HTTP-Version des Servers noch nicht kennt, gibt CICS vor dem Senden Ihrer Anforderung eine zusätzliche Anforderung aus, um die HTTP-Version des Servers zu ermitteln. Falls der Header 'Expect' nicht geeignet ist, sendet CICS Ihre Anforderung ohne ihn.

Diese Option darf nur verwendet werden, wenn Ihre Anforderung einen Nachrichtenhauptteil besitzt.

AUTHENTICATE(*cvda*)

Gibt Details über die Benutzerauthentifizierung an, damit der Zugriff auf eingeschränkte Daten gesteuert wird. Beim Einsatz von CICS als HTTP-Client gelten die folgenden CVDA-Werte:

NONE

Gibt an, dass keine Einschränkungen für den Zugriff auf diese Daten bestehen und daher keine Berechtigungsnachweise erforderlich sind. Dies ist der Standardwert für die Option AUTHENTICATE.

BASICAUTH

Gibt an, dass bei dieser Sitzung Berechtigungsnachweise für die HTTP-Basisauthentifizierung erforderlich sind. Diese Details können im Befehl selbst oder mithilfe des globalen Benutzerexits XWBAUTH angegeben werden.

CHANNEL(*datenwert*)

Gibt den Namen des Kanals an, zu dem der Container gehört. Der Name des Kanals kann aus bis zu 16 alphanumerischen Zeichen inklusive der entsprechenden Interpunktion bestehen. Führende und eingebettete Leerzeichen sind nicht zulässig. Falls der Name weniger als 16 Zeichen umfasst, wird er mit nachgestellten Leerzeichen aufgefüllt. Sie können den Kanalnamen DFHTRANSACTION angeben, um den Transaktionskanal zu verwenden.

Falls die Option CONTAINER angegeben ist, ist die Option CHANNEL optional.

Ist die Option CHANNEL nicht angegeben, geht CICS vom aktuellen Kanal aus.

CHARACTERSET(*datenwert*)

Gibt den Zeichensatz an, in den CICS den Entitätshauptteil der Anforderung vor dem Sendevorgang konvertiert. Der Name des Zeichensatzes kann aus bis zu 40 alphanumerischen Zeichen inklusive der entsprechenden Interpunktion bestehen. CICS unterstützt nicht alle von IANA genannten Zeichensätze. Im Abschnitt HTML-codierte Zeichensätze sind die IANA-Zeichensätze aufgelistet, die durch CICS für die Codepagekonvertierung unterstützt werden.

Damit eine Codepagekonvertierung des Entitätshauptteils stattfindet, muss die Option CLIENTCONV mit CLICONVERT angegeben sein bzw. den Standardwert verwenden können. Bei Angabe von NOCLICONVERT wird die Konvertierung des Entitätshauptteils unterdrückt. Wenn eine Konvertierung angefordert wird und das Attribut CHARACTERSET nicht angegeben ist, wird ISO-8859-1 als Standardwert verwendet.

CHUNKING(*cvda*)

Wird zum Steuern des Nachrichtensendevorgangs verwendet, wenn die Nachricht in Blöcke aufgeteilt gesendet wird (dies wird als 'Chunked Transfer Coding' bezeichnet). Wenn die Option nicht angegeben ist, wird das Chunked Transfer Coding standardmäßig nicht verwendet.

Der Inhalt einer in Blöcke aufgeteilten Nachricht kann auf eine Weise in Blöcke aufgeteilt werden, die für das Anwendungsprogramm geeignet ist. Der Hauptteil einer in Blöcke aufgeteilten Nachricht kann nicht direkt aus CICS-Dokumenten gebildet werden, weshalb die Option DOCTOKEN nicht verwendet werden kann.

Verwenden Sie für jeden Block der Nachricht einen separaten Befehl WEB SEND mit der Option CHUN-
KYES. Geben Sie mit der Option FROM den Datenblock und mit der Option FROMLENGTH die Länge
des Blocks an. Weitere Optionen für die Nachricht (z. B. die Option CLOSESTATUS) können im ersten
Befehl WEB SEND der Befehlsfolge (mit dem der erste Block gesendet wird) angegeben werden; ge-
ben Sie jedoch in nachfolgenden Befehlen (mit denen der zweite und darauffolgende Blöcke gesendet
werden) keine Optionen an.

Sobald Sie den letzten Datenblock gesendet haben, geben Sie einen weiteren Befehl WEB SEND mit
der Option CHUNKEND und ohne die Option FROM oder FROMLENGTH an. CICS sendet dann einen lee-
ren Block an den Empfänger, um die in Blöcke aufgeteilte Nachricht abzuschließen.

Falls Ihr Anwendungsprogramm im Rahmen des Chunking-Prozesses über einen Fehler benachrich-
tigt wird, können Sie mit dem Befehl WEB CLOSE den Prozess stoppen und die Verbindung schließen.
Der letzte leere Block wird vom Empfänger der in Blöcke aufgeteilten Nachricht nicht empfangen,
weshalb der Empfänger die bislang gesendeten Daten ignoriert und löscht.

Der Abschnitt HTTP-Anforderung oder -Antwort mit Chunked Transfer Coding senden enthält eine
umfassende Beschreibung der Prozedur für das Chunked Transfer Coding, die Sie befolgen müssen,
damit Ihre in Blöcke aufgeteilte Nachricht vom Empfänger angenommen werden kann.

Die CVDA-Werte lauten wie folgt:

CHUNKNO

Das Chunked Transfer Coding wird nicht für die Nachricht verwendet. CHUNKNO ist der Standard-
wert, wenn die Option CHUNKING nicht angegeben ist.

CHUNKYES

Das Chunked Transfer Coding wird verarbeitet. Die durch die Option FROM angegebenen Daten stel-
len einen Block der Nachricht dar.

CHUNKEND

Das Chunked Transfer Coding ist abgeschlossen. Für diesen Sendevorgang werden keine Daten
angegeben. CICS sendet einen leeren Block an den Empfänger, um die in Blöcke aufgeteilte Nach-
richt abzuschließen.

Anmerkungen:

1. Die Methode (angegebenen mit der Option METHOD) muss mit dem Chunked Transfer Coding
kompatibel sein.
2. Nachdem Sie mit dem Senden der Teile einer in Blöcke aufgeteilten Nachricht begonnen haben,
kann das Anwendungsprogramm keine anderen Nachrichten senden oder Elemente empfangen,
bis der abschließende leere Block gesendet wurde und die in Blöcke aufgeteilte Nachricht voll-
ständig ist.

Geben Sie bei Verwendung der Option CONTAINER die Option CHUNKING nicht an. Eine in Blöcke
aufgeteilte Antwort kann nicht aus einem Container gesendet werden.

CLOSESTATUS(*cvda*)

Gibt an, ob ein Verbindungsheader mit der Verbindungsoption 'close' (Connection: close) in die Nach-
richt einbezogen wird. Standardmäßig wird der Header nicht einbezogen. Die CVDA-Werte lauten wie
folgt:

CLOSE

Veranlasst CICS, für diese Anforderung einen Header 'Connection: close' zu schreiben. Der Header
benachrichtigt den Server, dass die Verbindung geschlossen wird, nachdem der Server seine Ant-

wort auf die Anforderung gesendet hat. (Bei einem Server mit Version HTTP/1.0, erzielt CICS dieselbe Wirkung, indem der Header 'Connection: Keep-Alive' weggelassen wird.) Geben Sie diese Option nicht an, falls Sie das Verbindungspooling in der URIMAP-Ressource für diese Verbindung implementiert haben, weil eine geschlossene Verbindung nicht für die Wiederverwendung in den Pool gestellt werden kann. Geben Sie diese Option ausschließlich dann an, wenn Sie die letzte Anforderung an den Server ausgeben und das Verbindungspooling nicht verwendet wird.

Wenn Sie die Option CLOSE in einem Befehl WEB SEND angeben, können keine weiteren Nachrichten an den Server gesendet werden, bis eine neue Verbindung hergestellt worden ist. Ausgenommen sind Fällen, bei denen das Chunked Transfer Coding verwendet wird; hier können Sie die Option CLOSE für den ersten Block der Nachricht angeben, um den Server darüber zu informieren, dass die Verbindung nach Abschluss der in Blöcke aufgeteilten Nachricht und dem Senden einer Antwort geschlossen wird.

NOCLOSE

Bedeutet, dass der Header 'Connection: close' nicht für diese Anforderung verwendet wird. Falls für den Server die Version HTTP/1.0 erkannt wird, sendet CICS einen Verbindungsheader mit der Verbindungsoption 'Keep-Alive' (Connection: Keep-Alive), um den Server zu benachrichtigen, dass eine persistente Verbindung erforderlich ist.

CLIENTCONV(*cvda*)

Gibt an, ob CICS den Entitätshauptteil der HTTP-Anforderung vor dem Sendevorgang aus der von der Anwendung verwendeten Codepage in einen für den Empfänger geeigneten Zeichensatz konvertiert. Falls diese Option nicht angegeben ist, wird der Entitätshauptteil standardmäßig konvertiert, sofern nicht ein anderer Medientyp als Text angegeben ist. Die CVDA-Werte lauten wie folgt:

CLICONVERT

CICS konvertiert den Entitätshauptteil der HTTP-Anforderung aus dem von der Anwendung verwendeten Codepage in den Zeichensatz, den Sie für den Server angeben. Mit der Option CHARACTERSET können Sie in diesem Befehl den verwendeten Zeichensatz angeben. Falls eine Konvertierung angefordert wird, Sie jedoch keinen Zeichensatz angeben, konvertiert CICS den Entitätshauptteil standardmäßig in den Zeichensatz ISO-8859-1. (Die von der Anwendung verwendete Codepage wurde bereits im Befehl WEB OPEN für die Verbindung angegeben.)

Bei anderen Medientypen als Text konvertiert CICS den Nachrichtenhauptteil nur unter den folgenden Umständen:

- Der Nachrichtenhauptteil wird mit der Option FROM aus einem Puffer gesendet und die Option CLICONVERT und/oder die Option CHARACTERSET ist angegeben.
- Der Nachrichtenhauptteil wird mit der Option DOCTOKEN aus einem Dokument gesendet.
- Der Nachrichtenhauptteil wird aus einem Container gesendet und die Option CHARACTERSET ist angegeben.

Container mit dem Datentyp BIT enthalten keine Textdaten und unterstützen daher keine Codepagekonvertierung. Infolgedessen wird ein Fehler INVREQ mit einem RESP2-Wert erzeugt, wenn Sie eine der Optionen CLICONVERT oder CHARACTERSET mit einem Container des Datentyps BIT codieren.

NOCLICONVERT

CICS konvertiert den Entitätshauptteil der HTTP-Anforderung nicht; dieser wird in dem von der Anwendung verwendeten Zeichensatz an den Server gesendet, der im Befehl WEB OPEN für die Verbindung angegeben war.

CONTAINER(*datenwert*)

Gibt den Namen des Containers an, in dem der HTTP-Hauptteil aufbewahrt wird, bevor er an den Server gesendet wird. Der Name des Containers kann aus bis zu 16 alphanumerischen Zeichen inklusive der entsprechenden Interpunktion bestehen. Führende und eingebettete Leerzeichen sind nicht zulässig. Falls der Name weniger als 16 Zeichen umfasst, wird er mit nachgestellten Leerzeichen aufgefüllt.

Bei Angabe der Option CONTAINER speichert CICS den HTTP-Hauptteil im 64-Bit-Speicher und sendet ihn von dort aus.

DOCSTATUS(*cvda*)

Gibt an, ob das Dokument während der Verarbeitung des Befehls WEB SEND gelöscht wird oder nicht. Die CVDA-Werte lauten wie folgt:

DOCDELETE

CICS löscht das Dokument, nachdem der Dokumentinhalt für den Sendevorgang gespeichert wurde. Der dem Dokument zugeordnete Speicher wird sofort freigegeben. Falls Sie nachfolgende Anforderungen für das Dokument ausgeben, generieren diese Anforderungen eine Antwort TOKENERR.

NODOCDELETE

CICS löscht das Dokument während der Verarbeitung des Befehls WEB SEND nicht. Dies ist der Standardwert für die Option DOCSTATUS.

DOCTOKEN(*datenwert*)

Gibt das 16 Byte umfassende binäre Token eines Dokuments an, das als Nachrichtenhauptteil gesendet werden soll. Das Dokument erstellen Sie mit der CICS-Dokumentschnittstelle (EXEC CICS-Befehle DOCUMENT CREATE, INSERT und SET). Sie müssen das Dokument vor dem Sendevorgang nicht abrufen. Als alternatives Verfahren für die Erstellung eines Nachrichtenhauptteils steht die Option FROM zur Verfügung.

Der Hauptteil einer in Blöcke aufgeteilten Nachricht kann nicht aus CICS-Dokumenten gebildet werden, weshalb die Option DOCTOKEN für das Chunked Transfer Coding nicht verwendet werden kann.

Bei Angabe der Option DOCTOKEN wird das Dokument im 31-Bit-Speicher verarbeitet.

FROM(*datenbereich*)

Gibt einen Puffer von Daten an, in dem sich der Nachrichtenhauptteil befindet. Der Nachrichtenhauptteil wird durch das Anwendungsprogramm erstellt. Wenn Sie die Option FROM angeben, müssen Sie mit der Option FROMLENGTH die Länge des Datenpuffers angeben. Als alternatives Verfahren für die Erstellung eines Nachrichtenhauptteils stehen die Optionen DOCTOKEN und CONTAINER zur Verfügung; die Option DOCTOKEN kann jedoch nicht für den Hauptteil einer in Blöcke aufgeteilten Nachricht verwendet werden.

Für die Größe des Datenbereichs gibt es keinen Maximalwert; aufgrund von Speicheraspekten ist seine Größe in der Praxis jedoch begrenzt. Der Abschnitt Entitätshauptteil für eine HTTP-Nachricht erstellen enthält weitere Informationen zu Speicheraspekten.

Wenn die Option FROM angegeben ist und CICS den HTTP-Hauptteil vor dem Senden kopieren oder konvertieren muss, wird der Hauptteil im 64-Bit-Speicher kopiert oder konvertiert und von dort aus gesendet.

FROMLENGTH(*datenwert*)

Gibt als Vollwort-Binärwert die Länge des Datenpuffers an, der in der Option FROM (für den Nachrichtenhauptteil) angegeben ist. Dieser Wert muss korrekt angegeben werden, weil eine falsche Datengröße für den Empfänger der Nachricht zu Problemen führen kann.

MEDIATYPE(*datenwert*)

Gibt den Dateninhalt eines bereitgestellten Nachrichtenhauptteils an, z. B. text/xml. Als Wert für die Option MEDIATYPE müssen Sie einen 56 Byte umfassenden Bereich angeben. Der Medientyp kann bis zu 56 alphanumerische Zeichen lang sein, inklusive der entsprechenden Interpunktion, jedoch ohne Leerzeichen. Weitere Informationen zu Medientypen finden Sie unter IANA-Datenträgertypen und -Zeichensätze. CICS überprüft, ob das Format des Medientyps richtig ist, nimmt jedoch keine Überprüfung für die Gültigkeit des Medientyps in Bezug auf den Dateninhalt vor. CICS erstellt anhand dieser Informationen den Header 'Content-Type' für die Nachricht.

Bei Anforderungen, die einen Hauptteil erfordern, müssen Sie die Option MEDIATYPE angeben. Es gibt keinen Standardwert. Falls der erforderliche Header 'Content-Type' jedoch Leerzeichen oder mehr als 56 Zeichen enthalten muss, kann die Anwendung ihn mit dem Befehl WEB WRITE HTTPHEADER bereitstellen. Geben Sie in diesem Fall die Option MEDIATYPE nicht an.

Anhand des bereitgestellten Medientyps wird unter den folgenden Umständen ermittelt, ob eine Copagekonvertierung erforderlich ist:

- Sie senden mit der Option FROM eine Nachricht aus einem Puffer und die Optionen CLIENTCONV und CHARACTERSET sind nicht angegeben.
- Sie senden mit der Option DOCTOKEN eine Nachricht aus einem Dokument und die Optionen CLIENTCONV und CHARACTERSET sind nicht angegeben.
- Sie senden mit der Option CONTAINER eine Nachricht aus einem benannten Container und entweder ist die Option CLICONVERT angegeben oder die Optionen CLIENTCONV und CHARACTERSET sind nicht angegeben.

Falls Text als Medientyp angegeben ist, wird die Nachricht konvertiert. Ist nicht Text als Medientyp angegeben, wird die Nachricht nicht konvertiert.

METHOD(*cvda*)

Gibt die HTTP-Methode für die Anforderung an.

Von diesem Befehl werden die Methoden GET, HEAD, PATCH, POST, PUT, TRACE, OPTIONS und DELETE unterstützt. Manche HTTP-Server, insbesondere Server mit der Version HTTP/1.0, implementieren jedoch möglicherweise nicht alle dieser Methoden.

Der Abschnitt HTTP-Methodenreferenz für CICS-Webunterstützung enthält weitere Informationen zur korrekten Verwendung von Methoden, inklusive der HTTP-Versionen, die für die einzelnen Methoden gelten.

CICS blockiert das Senden eines Nachrichtenhauptteils für Methoden, bei denen er ungeeignet ist, und macht ihn für Methoden erforderlich, bei denen er geeignet ist. Bei Methoden, die keinen Anforderungshauptteil besitzen, ist das Chunked Transfer Coding irrelevant.

Die CVDA-Werte lauten wie folgt:

GET

Fordert eine Ressource vom Server an. Ein Anforderungshauptteil ist nicht zulässig.

HEAD

Fordert für eine Ressource die HTTP-Header, jedoch nicht den Antworthauptteil an. Ein Anforderungshauptteil ist nicht zulässig.

PATCH

Fordert an, dass eine Gruppe von Änderungen auf die durch den Anforderungs-URI identifizierte Ressource angewendet wird. Ein Anforderungshauptteil ist erforderlich.

POST

Sendet Daten an einen Server. Ein Anforderungshauptteil ist erforderlich.

PUT

Erstellt oder ändert eine Ressource auf dem Server. Ein Anforderungshauptteil ist erforderlich.

TRACE

Erstellt einen Trace für die Route Ihrer Anforderung zum Server. Ein Anforderungshauptteil ist nicht zulässig.

OPTIONS

Ruft Informationen zum Server ab. Ein Anforderungshauptteil ist zulässig, besitzt jedoch keinen definierten Zweck. Falls Sie einen Anforderungshauptteil verwenden, müssen Sie einen Medientyp angeben.

DELETE

Löscht eine Ressource auf dem Server. Ein Anforderungshauptteil ist nicht zulässig.

PASSWORD(*datenwert*)

Gibt das Kennwort an, das der Benutzer-ID oder dem Anmeldenamen zugeordnet ist, die/der auf diese Daten zugreifen darf. Die Option PASSWORD ist nur erforderlich, wenn die Option USERNAME verwendet wird.

Falls Sie die Optionen USERNAME und PASSWORD im Befehl **WEB SEND** und ebenfalls die Option AUTHENTICATE in der URIMAP-Ressource angeben, werden die Werte aus dem Befehl WEB SEND verwendet.

Falls das angegebene Kennwort länger als 8 Zeichen ist, wird es als Kennwortphrase behandelt, wenn es an z/OS-Systeme gesendet wird.

PASSWORDLEN(datenwert)

Gibt als Vollwort-Binärzahlvariable die Länge des für die Option PASSWORD angegebenen Puffers an.

PATH(datenbereich)

Gibt die Pfadinformationen für die spezielle Ressource im Server an, auf die die Anwendung Zugriff benötigt.

Falls mit der Option URIMAP im Befehl WEB OPEN für diese Verbindung eine vorhandene URIMAP-Definition angegeben wurde, ist der in der URIMAP-Definition festgelegte Pfad der Standardpfad für den Befehl WEB SEND. Falls Sie unter diesen Umständen keine Pfadinformationen im Befehl WEB SEND angeben, wird der Pfad aus der URIMAP-Definition verwendet. Falls Sie einen anderen Pfad angeben, als in der URIMAP-Definition festgelegt ist, überschreibt dieser Wert den Pfad aus der URIMAP-Definition.

Für den Fall, dass die Option URIMAP im Befehl WEB OPEN nicht verwendet wurde, gibt es keinen Standardpfad und Sie müssen Pfadinformationen angeben. Pfadinformationen können mit dem Befehl WEB PARSE URL aus einer bekannten URL extrahiert werden.

Als Alternative zur Angabe der Pfadinformationen mit der Option PATH können Sie auch mit der Option URIMAP des Befehls WEB SEND eine URIMAP-Definition angeben, aus der die Pfadinformationen direkt entnommen werden.

PATHLENGTH(datenwert)

Gibt als Vollwort-Binärwert die Länge des Pfades an. Falls Sie mit der Option PATH Pfadinformationen bereitstellen, müssen Sie die Option PATHLENGTH angeben. Die Informationen zur Pfadlänge werden zurückgegeben, wenn Sie eine URL mit dem Befehl WEB PARSE URL analysieren.

QUERYSTRING(datenbereich)

Gibt eine Abfragezeichenfolge an, die dem Server als Teil der Anforderung übergeben werden soll. Die Angabe eines Fragezeichens (?) am Beginn der Abfragezeichenfolge ist nicht zwingend erforderlich; falls Sie das Fragezeichen nicht angeben, stellt CICS es beim Erstellen der Anforderung automatisch für Sie bereit. Wenn Sie Zeichen mit Escapezeichen in die Abfragezeichenfolge einbeziehen, übergibt CICS die Zeichen in ihrem Format nach Anwendung der Escapezeichen an den Server.

QUERYSTRLEN(datenwert)

Gibt als Vollwort-Binärwert die Länge der in der Option QUERYSTRING bereitgestellten Abfragezeichenfolge an.

SESTOKEN(datenwert)

Gibt den 8 Byte umfassenden Binärwert für das Sitzungstoken an, das eine Verbindung zwischen CICS und einem Server eindeutig kennzeichnet. Dieser Wert wird durch einen Befehl WEB OPEN für CICS als HTTP-Client zurückgegeben. Die Verwendung des Sitzungstokens ist im Abschnitt [Sitzungstokens](#) erläutert.

URIMAP(datenwert)

Gibt den Namen ,bis zu 8 Zeichen in Groß-/Kleinschreibung, einer URIMAP-Definition an, die Pfadinformationen für die jeweilige Ressource im Server bereitstellt, auf die die Anwendung zugreift. Die URIMAP-Definition muss für CICS als HTTP-Client, mit Angabe von USAGE(CLIENT), bestimmt sein. Ihr Attribut HOST muss mit dem Attribut HOST der URIMAP-Definition, die im Befehl WEB OPEN für diese Verbindung angegeben wurde, oder mit dem Hostnamen identisch sein, der in der Option HOST des Befehls WEB OPEN für diese Verbindung angegeben wurde. Eine im Befehl WEB SEND angegebene URIMAP-Definition gilt nur für diese Anforderung.

Geben Sie die Option PATH oder PATHLENGTH nicht an, falls die Option URIMAP angegeben wird.

USERNAME(datenwert)

Gibt die Benutzer-ID oder den Anmeldenamen an, die/der auf diese Daten zugreifen dar. Bei Angabe der Option USERNAME müssen Sie auch die Option PASSWORD verwenden.

Falls Sie die Optionen USERNAME und PASSWORD im Befehl **WEB SEND** und ebenfalls die Option AUTHENTICATE in der URIMAP-Ressource angeben, werden die Werte aus dem Befehl WEB SEND verwendet.

USERNAMELEN(*datenwert*)

Gibt als Vollwort-Binärzahlvariable die Länge des für die Option USERNAME angegebenen Puffers an.

Bedingungen**122 CHANNELERR**

Die RESP2-Werte lauten wie folgt:

2

Der in der Option CHANNEL angegebene Kanal wurde nicht gefunden.

110 CONTAINERERR

Die RESP2-Werte lauten wie folgt:

2

Der in der Option CONTAINER angegebene Container wurde nicht gefunden.

16 INVREQ

Die RESP2-Werte lauten wie folgt:

11

Der Aktionscode ist ungültig.

12

Es sind beide Optionen URIMAP und PATH angegeben. Es ist nur eine einzige Option zulässig bzw. die Option URIMAP ist nicht für den zweiten Block und nachfolgende Blöcke zulässig.

13

Der Status für den Schließvorgang ist ungültig.

15

Bei der Codepagekonvertierung trat ein Fehler auf.

17

Eine Anforderung mit 'Expect-100' wurde durch den Server zurückgewiesen.

22

Die Datenblockgröße ist ungültig.

32

Der Medientyp ist ungültig.

33

Die Methode unterstützt keinen Hauptteil.

34

Die Methode erfordert einen Hauptteil.

43

Der angegebene Wert für DOCSTATUS ist ungültig.

45

Der angegebene Zeichensatz ist ungültig.

46

Die Option CLIENTCONV ist ungültig.

49

Das Format der Pfadoptio ist ungültig.

54

Die HTTP-Methode ist ungültig.

63

Das URIMAP-Objekt wurde inaktiviert.

64

Der Host in der URIMAP-Definition stimmt nicht mit dem Host überein, der beim Öffnen dieser Sitzung angegeben wurde.

- 69** Das Chunked Transfer Coding wird bei dieser HTTP-Version nicht unterstützt.
- 71** Beim Chunked Transfer Coding trat ein Fehler auf.
- 74** Die Verbindung wurde geschlossen. Der Server hat möglicherweise aufgrund von Inaktivität über diese Verbindung ein zulässiges Zeitlimit überschritten.
- 76** Die Option MEDIATYPE ist erforderlich.
- 79** Das Pipelining wird gegenwärtig durchgeführt. Der Header 'Expect' kann nicht gesendet werden.
- 80** Die Option CHARACTERSET kann nicht zusammen mit NOCLICONVERT angegeben werden.
- 120** Der Wert für die Option CHUNKING ist ungültig.
- 121** Die Option FROMLENGTH ist erforderlich.
- 122** Die Option FROM ist erforderlich.
- 123** Es wurde kein Nachrichtenhauptteil angegeben. Verwenden Sie die Option FROM, DOCTOKEN oder CHUNKEND.
- 124** Die Option CHUNKING war nicht angegeben, die Option FROMLENGTH ist erforderlich.
- 125** Die Option CHUNKNO war angegeben, die Option FROM ist erforderlich.
- 126** Die Option CHUNKNO war angegeben, die Option FROMLENGTH ist erforderlich.
- 127** Die Option CHUNKYES war angegeben, die Option FROM ist erforderlich.
- 128** Die Option CHUNKYES war angegeben, die Option FROMLENGTH ist erforderlich.
- 129** Die Option FROM ist bei Verwendung der Option CHUNKEND nicht zulässig.
- 130** Die Option FROMLENGTH ist bei Verwendung der Option CHUNKEND nicht zulässig.
- 131** Für die Option FROMLENGTH war Null angegeben.
- 132** Die Option METHOD ist für den zweiten und alle nachfolgenden Blöcke nicht zulässig.
- 133** Die Option MEDIATYPE ist für den zweiten und alle nachfolgenden Blöcke nicht zulässig.
- 135** Die Option PATH ist für den zweiten und alle nachfolgenden Blöcke nicht zulässig.
- 136** Die Option METHOD ist erforderlich.
- 142** Der Wert für AUTHENTICATE ist ungültig. Der CVDA-Wert lautet NONE oder BASICAUTH.
- 144** Einer oder mehrere Parameter des Webbefehls sind ungültig.

145

Der Kanal wurde nicht angegeben und es gibt keinen aktuellen Kanal.

147

Es trat ein interner Konvertierungsfehler auf.

150

Die Konvertierung wurde angefordert, aber die zu sendenden Daten befinden sich in einem Container mit dem Datentyp BIT.

17 IOERR

Die RESP2-Werte lauten wie folgt:

42

Es trat ein Socketfehler auf.

22 LENGERR

Die RESP2-Werte lauten wie folgt:

5

Der Wert für die Option PATHLENGTH war nicht größer als Null.

8

Der Wert für die Option QUERYSTRLEN war nicht größer als Null.

50

Der Wert für die Option FROMLENGTH war nicht größer als Null.

139

Der Wert für USERNAMELEN ist negativ oder größer als 256.

140

Der Wert für PASSWORDLEN ist negativ oder größer als 256.

70 NOTAUTH

Die RESP2-Werte lauten wie folgt:

100

Der Pfad wurde durch den Sicherheitsexit gesperrt.

110

Es trat ein Fehler für XWBAUTH auf. Der globale Benutzerexit XWBAUTH hat einen Rückgabecode UERCERR ausgegeben, weil der Exit XWBAUTH erforderlich ist, jedoch keine gültige Antwort zurückgeben kann.

Dieser Fehlercode wird ausgegeben, wenn Folgendes zutrifft: BASICAUTH ist angegeben; USERNAME und/oder PASSWORD ist nicht angegeben; XWBAUTH ist inaktiv oder gibt die Antwort UERCERR zurück.

13 NOTFND

Die RESP2-Werte lauten wie folgt:

61

Das angegebene URIMAP-Objekt wurde nicht gefunden.

19 NOTOPEN

Die RESP2-Werte lauten wie folgt:

27

Das Sitzungstoken ist ungültig.

124 TIMEDOUT**156**

Beim Senden des Sockets trat eine Zeitlimitüberschreitung auf.

112 TOKENERR

Die RESP2-Werte lauten wie folgt:

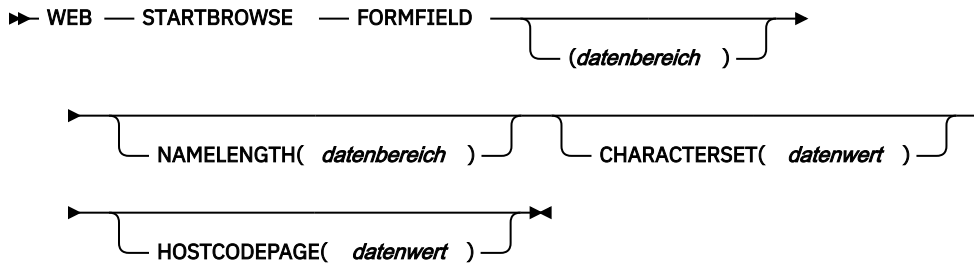
47

Das angegebene Dokumenttoken ist ungültig oder das Dokument wurde gelöscht.

WEB STARTBROWSE FORMFIELD

Signalisiert den Start eines Suchvorgangs in einem HTML-Formularfeld.

WEB STARTBROWSE FORMFIELD



Bedingungen: INVREQ, LENGERR, NOTFND

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl `WEB STARTBROWSE FORMFIELD` macht den Beginn eines Suchvorgangs nach einer Gruppe von Name/Wert-Paaren in einem HTML-Formular kenntlich, das zum Hauptteil einer HTTP-Anforderung gehört, die durch die aktuelle CICS-Task verarbeitet wird.

Optionen

CHARACTERSET(name)

Gibt den 40 Zeichen langen Namen des Zeichensatzes an, der zur Codierung der Formulardaten erforderlich ist. Diese Option sollte mit der Formularcodierung übereinstimmen, die durch das entsprechende HTML-Formular festgelegt ist (weitere Informationen finden Sie unter Vorgehensweise zur Bestimmung der Clientcodierung). CICS unterstützt nicht alle von IANA genannten Zeichensätze. Im Abschnitt HTML-codierte Zeichensätze sind die IANA-Zeichensätze aufgelistet, die durch CICS für die Codepagekonvertierung unterstützt werden.

CLNTCODEPAGE(name)

Diese Option wird nur zu Upgradezwecken unterstützt. Sie wurde durch die Option `CHARACTERSET` ersetzt. Bei beiden Schlüsselwörtern führt CICS dieselbe Aktion aus.

FORMFIELD(datenbereich)

Ist das Schlüsselwort, das den Befehl `STARTBROWSE FORMFIELD` einleitet. Sie können optional den Namen des Formularfeldes angeben, bei dem der Suchvorgang beginnen soll, indem Sie diesen Namen beispielsweise in einem Datenbereich angeben, auf den die Option `NAMELENGTH` folgt.

```
WEB STARTBROWSE FORMFIELD(name) NAMELENGTH(länge)
```

Der Name ist eine Textzeichenfolge, die den Namen des angeforderten Feldes enthält. Falls kein Name angegeben wird, beginnt der Suchvorgang beim ersten Name/Wert-Paar im HTML-Formular.

HOSTCODEPAGE(name)

Gibt den 8 Zeichen langen Namen der CICS-Codepage (für den Host) an, die durch das Anwendungsprogramm erforderlich ist und in die die Formulardaten konvertiert werden sollen. Normalerweise handelt es sich hierbei um eine EBCDIC-Codepage.

Das CICS-Standardformat für einen Codepagenamen besteht aus der Codepagenummer (oder allgemeiner der CCSID), die nach Bedarf mithilfe von 3 bis 5 Dezimalziffern geschrieben und anschließend mit nachgestellten Leerzeichen auf 8 Zeichen aufgefüllt wird. Bei der Codepage 37, die aus weniger als 3 Ziffern besteht, lautet das Standardformat '037'. CICS akzeptiert jetzt als Codepagenamen auch eine beliebige Dezimalzahl mit bis zu 8 Ziffern (aufgefüllt mit nachgestellten Leerzeichen) im Bereich von 1 bis 65535, auch wenn dieser nicht im Standardformat angegeben ist.

Falls die Codepage nicht angegeben ist, werden die Daten in der EBCDIC-Codepage zurückgegeben, die im Systeminitialisierungsparameter `LOCALCCSID` angegeben ist (der für die lokale CICS-Region gilt).

und den Standardwert 037 besitzt), unter der Voraussetzung, dass die angegebene Codepage durch die CICS-Webschnittstelle unterstützt wird. Die Codepage wird unterstützt, wenn sie in einer Liste von EBCDIC-Codepages angegeben ist, die von CICS als ausreichender Standard für eine erfolgreiche Analyse von Web-Headern anerkannt werden (hierzu zählen alle SBCS-CECP und Euro-Codepages). Andernfalls gibt CICS die Daten stattdessen in der EBCDIC-Standardcodepage 037 zurück.

NAMELENGTH(datenwert)

Gibt als Vollwort-Binärwert die Länge des Formularfeldnamens an. Falls mit der Option FORMFIELD ein Datenbereich für den Namen angegeben wird, muss dieses Feld angegeben sein.

Bedingungen

21 ILLOGIC

Der RESP2-Wert lautet wie folgt:

5

Es wird bereits ein Suchvorgang für Formularfelder ausgeführt.

16 INVREQ

Tritt bei den folgenden Bedingungen auf. Die RESP2-Werte lauten wie folgt:

1

Der Befehl wurde in einer Anwendung ausgegeben, die nicht zur CICS-Webunterstützung gehört.

3

Der Befehl wurde für eine Anforderung ausgegeben, die keine HTTP-Anforderung ist.

11

Die Client-Codepage wurde nicht gefunden.

12

Die Host-Codepage wurde nicht gefunden.

13

In der HTTP-Anforderung wurden keine Formulardaten übergeben.

14

Die Kombination der Codepages für den Client und den Server ist ungültig.

17

In der Eingabenachricht wurden ungültige Formulardaten gefunden.

153

Der Formulartyp ist unbekannt.

154

In den Formulardaten wurde eine Begrenzungszeichenfolge erwartet, jedoch nicht gefunden.

22 LENGERR

Tritt bei den folgenden Bedingungen auf. Die RESP2-Werte lauten wie folgt:

1

Der Wert in der Option NAMELENGTH oder VALUELENGTH ist kleiner-gleich Null.

13 NOTFND

Tritt bei den folgenden Bedingungen auf. Die RESP2-Werte lauten wie folgt:

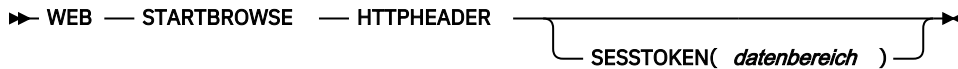
1

Der im Parameter FORMFIELD angegebene Formularfeldname wurde nicht gefunden.

WEB STARTBROWSE HTTPHEADER

Signalisiert den Start eines Suchvorgangs in einem HTTP-Header.

WEB STARTBROWSE HTTPHEADER



Bedingungen: ILLOGIC, INVREQ, NOTFND, NOTOPEN

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl **WEB STARTBROWSE HTTPHEADER** signalisiert den Beginn eines Suchvorgangs nach HTTP-Headerinformationen. Die Option **SESSTOKEN** ist erforderlich, falls die HTTP-Headerinformationen Teil einer Antwort sind, die an CICS als HTTP-Client gesendet wird.

Optionen

SESSTOKEN(datenwert)

Wenn CICS als HTTP-Client eingesetzt wird, ist diese Option erforderlich. Sie gibt den 8 Byte umfassenden Binärwert für das Sitzungstoken an, das eine Verbindung zwischen CICS und einem Server eindeutig kennzeichnet. Dieser Wert wird durch einen Befehl **WEB OPEN** für CICS als HTTP-Client zurückgegeben. Die Verwendung des Sitzungstokens ist im Abschnitt [Sitzungstokens](#) erläutert.

Bedingungen

21 ILLOGIC

Der RESP2-Wert lautet wie folgt:

10

Es wird bereits ein Suchvorgang für HTTP-Header ausgeführt.

16 INVREQ

Die RESP2-Werte lauten wie folgt:

1

Der Befehl wurde in einer Anwendung ausgegeben, die nicht zur CICS-Webunterstützung gehört.

3

Der Befehl wurde für eine Anforderung ausgegeben, die keine HTTP-Anforderung ist.

43

Es wurden keine HTTP-Header gefunden.

13 NOTFND

Der RESP2-Wert lautet wie folgt:

1

Der Header wurde nicht gefunden.

19 NOTOPEN

Der RESP2-Wert lautet wie folgt:

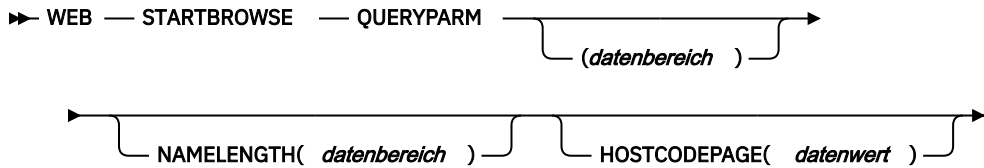
27

Das Sitzungstoken ist ungültig.

WEB STARTBROWSE QUERYPARM

Startet die Suche von Abfragezeichenfolgedaten in einer URL.

WEB STARTBROWSE QUERYPARM



Bedingungen: ILLOGIC, INVREQ, LENGERR, NOTFND

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl WEB STARTBROWSE QUERYPARM signalisiert den Beginn eines Suchvorgangs nach den aus Name/Wert-Paaren bestehenden Schlüsselwortparametern in der Abfragezeichenfolge einer URL.

Für Formulare können Sie auch den Befehl `WEB STARTBROWSE FORMFIELD` verwenden. Formulare sind Nachrichten mit den Medientypen `'application/x-www-form-urlencoded'` oder `'multipart/form-data'`.

Optionen

QUERYPARM(*datenbereich*)

Das Schlüsselwort, das den Befehl STARTBROWSE QUERYPARM einleitet. Sie können optional den Namen des Schlüsselwortparameters angeben, bei dem der Suchvorgang beginnen soll, indem Sie diesen Namen beispielsweise in einem Datenbereich angeben, auf den die Option NAMELENGTH folgt.

```
WEB STARTBROWSE QUERYPARM(name) NAMELENGTH(länge)
```

Der Name ist eine Textzeichenfolge, die den Namen des angeforderten Schlüsselwortparameters enthält. Falls kein Name angegeben wird, beginnt der Suchvorgang beim ersten Name/Wert-Paar in der Abfragezeichenfolge.

HOSTCODEPAGE(*datenwert*)

Gibt den 8 Zeichen langen Namen der CICS-Codepage (für den Host) an, die durch das Anwendungsprogramm erforderlich ist und in die die Daten der Abfragezeichenfolge konvertiert werden sollen. Normalerweise handelt es sich hierbei um eine EBCDIC-Codepage.

Das CICS-Standardformat für einen Codepagenamen besteht aus der Codepagenummer (oder allgemeiner der CCSID), die nach Bedarf mithilfe von 3 bis 5 Dezimalziffern geschrieben und anschließend mit nachgestellten Leerzeichen auf 8 Zeichen aufgefüllt wird. Bei der Codepage 37, die aus weniger als 3 Ziffern besteht, lautet das Standardformat '037'. CICS akzeptiert jetzt als Codepagenamen auch eine beliebige Dezimalzahl mit bis zu 8 Ziffern (aufgefüllt mit nachgestellten Leerzeichen) im Bereich von 1 bis 65535, auch wenn dieser nicht im Standardformat angegeben ist.

Falls die Codepage nicht angegeben ist, werden die Daten in der EBCDIC-Codepage zurückgegeben, die im Systeminitialisierungsparameter LOCALCCSID angegeben ist (der für die lokale CICS-Region gilt und den Standardwert 037 besitzt), unter der Voraussetzung, dass die angegebene Codepage durch die CICS-Webschnittstelle unterstützt wird. Die Codepage wird unterstützt, wenn sie in einer Liste von EBCDIC-Codepages angegeben ist, die von CICS als ausreichender Standard für eine erfolgreiche Analyse von Web-Headern anerkannt werden (hierzu zählen alle SBCS-CECP und Euro-Codepages). Andernfalls gibt CICS die Daten stattdessen in der EBCDIC-Standardcodepage 037 zurück.

NAMELENGTH(*datenwert*)

Gibt als Vollwort-Binärwert die Länge des Schlüsselwortparameternamens an. Falls mit der Option QUERYPARM ein Datenbereich für den Namen angegeben wird, muss dieses Feld angegeben sein.

Bedingungen

21 ILLOGIC

Der RESP2-Wert lautet wie folgt:

5

Es wird bereits ein Suchvorgang für Schlüsselwortparameter ausgeführt.

16 INVREQ

Die RESP2-Werte lauten wie folgt:

1

Der Befehl wurde in einer Anwendung ausgegeben, die nicht zur CICS-Webunterstützung gehört.

3

Der Befehl wurde für eine Anforderung ausgegeben, die keine HTTP-Anforderung ist.

12

Die Host-Codepage wurde nicht gefunden.

13

Es wurden keine Schlüsselwortparameter gefunden.

14

Die Kombination der Codepages für den Client und den Server ist ungültig.

17

In der HTTP-Anforderung wurden ungültige Schlüsselwortparameter gefunden.

22 LENGERR

Der RESP2-Wert lautet wie folgt:

1

Für den Parameter NAMELEN wurde ein ungültiger Wert angegeben.

13 NOTFND

Der RESP2-Wert lautet wie folgt:

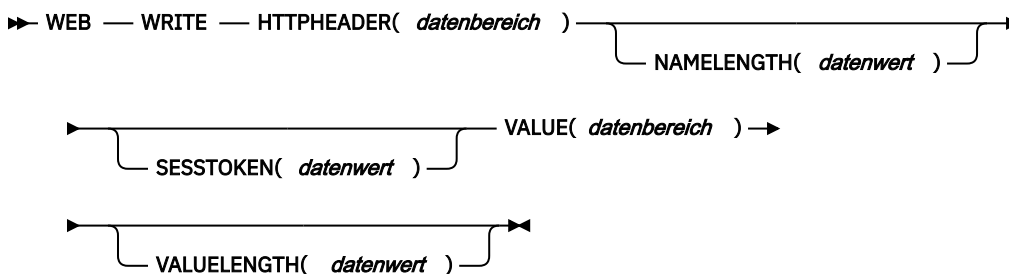
1

Der Schlüsselwortparameter wurde nicht gefunden.

WEB WRITE HTTPHEADER

Erstellt Informationen eines HTTP-Headers.

WEB WRITE HTTPHEADER



Bedingungen: INVREQ, LENGERR, NOTOPEN

Dieser Befehl ist threadsicher.

Beschreibung

Mit dem Befehl **WEB WRITE HTTPHEADER** kann eine Anwendung HTTP-Headerinformationen zu einer Nachricht hinzufügen. Wenn CICS als HTTP-Server verwendet wird, ist die Nachricht eine für einen Web-Client bestimmte Antwort. Wenn CICS als HTTP-Client eingesetzt wird, ist die Nachricht für einen Server bestimmte Anforderung und die Option **SESSTOKEN** ist angegeben.

Manche HTTP-Header werden durch CICS automatisch erstellt, falls sie für die Anwendung erforderlich sind; die Anwendung muss solche Header nicht selbst schreiben. Hierzu gehören die folgenden Header:

- Header für den ARM-Korrelator

- Header 'Connection'
- Header 'Content-Type' (wird durch CICS geschrieben, kann jedoch durch eine Clientanwendung bereitgestellt werden, falls ein komplexer Header erforderlich ist)
- Header 'Content-Length'
- Header 'Date'
- Header 'Expect'
- Header 'Host'
- Header 'Server' (die automatische Erstellung ist vom Systeminitialisierungsparameter **HTTPSERVERHDR** abhängig)
- Header 'TE' (wird durch CICS geschrieben, weitere Instanzen können jedoch hinzugefügt werden)
- Header 'Transfer-Encoding'
- Header 'User-Agent' (die automatische Erstellung ist vom Systeminitialisierungsparameter **HTTPUSRA-GENTHDR** abhängig)
- Header 'WWW-Authenticate'

Im Abschnitt HTTP-Headerreferenz für CICS-Webunterstützung sind die Umstände beschrieben, unter denen diese Header erstellt werden. Falls das Benutzeranwendungsprogramm einen Header schreibt, der ebenfalls von CICS generiert wird, richtet sich die Verarbeitung durch CICS nach der jeweiligen Situation:

- Wenn CICS als HTTP-Server eingesetzt wird und der Header für eine Antwort geeignet ist, überschreibt CICS den Header nicht, sondern lässt die Verwendung der Headerversion aus der Anwendung zu.
- Wenn CICS als HTTP-Client eingesetzt wird und der Header für eine Anforderung geeignet ist, lässt CICS nicht zu, dass der Header durch die Anwendung geschrieben wird und gibt eine Fehlerantwort an den Befehl **WEB WRITE HTTPHEADER** zurück. Ausgenommen hiervon sind die Header 'TE' und 'Content-Type'. Anwendungsprogramme können weitere Instanzen des Headers 'TE' hinzufügen. Anwendungsprogramme können ebenfalls den Header 'Content-Type' bereitstellen, falls der erforderliche Header Leerzeichen oder mehr als 56 Zeichen enthalten muss und daher nicht in der Option **MEDIATYPE** des Befehls **WEB SEND** angegeben werden kann.
- Falls der Header normalerweise nicht für den Nachrichtentyp (Anforderung oder Antwort) geeignet ist, lässt CICS den Header wie alle benutzerdefinierten Header zu. Diese Situation sollte nicht auftreten, wenn Ihre Nachricht mit der von Ihnen verwendeten HTTP-Spezifikation konform ist.
- Falls mit dem Befehl **WEB WRITE HTTPHEADER** ein Header 'Server' geschrieben wird, wird selbst dann ein Header 'Server' bereitgestellt, wenn der von CICS zur Verfügung gestellte Header 'Server' durch den Parameter **HTTPSERVERHDR=NO** unterdrückt wird.

Der Befehl **WEB WRITE HTTPHEADER** fügt einen einzigen Header hinzu; Sie können den Befehl wiederholen, wenn Sie weitere Header hinzufügen wollen. Falls Sie einen Header schreiben, den Sie bereits für die Anforderung bzw. Antwort geschrieben haben, fügt CICS den neuen Header zusätzlich zum vorhandenen Header zur Anforderung bzw. Antwort hinzu.

Der Name und Wert des von Ihnen geschriebenen Headers sowie die Umstände, unter denen Sie sich für das Schreiben eines Headers entscheiden, sollten die Anforderungen der von Ihnen verwendeten HTTP-Spezifikation berücksichtigen.

Falls Sie einen Header verwenden wollen, um eine Aktion anzufordern, die von einem Server oder Client mit einer Version vor HTTP/1.1 möglicherweise nicht ordnungsgemäß ausgeführt wird, und Sie sich vergewissern wollen, ob die Aktion erfolgreich ausgeführt werden kann, überprüfen Sie die HTTP-Version des Servers mit dem Befehl **WEB EXTRACT** unter Angabe der Option **HTTPVERSION**.

Wenn Sie CICS als HTTP-Client einsetzen, in der ersten Anforderung an den Server einen Header 'Trailer' (für eine in Blöcke aufgeteilte Nachricht) schreiben und die Optionen **HTTPVNUM** und **HTTPRNUM** im Befehl **WEB OPEN** für die Sitzung nicht angegeben waren, gibt CICS eine Anforderung mit der Methode **OPTIONS** aus, um die HTTP-Version des Servers zu überprüfen. Diese zusätzliche Anforderung wird nur für den Header 'Trailer' ausgegeben.

Der Befehl **WEB WRITE HTTPHEADER** kann nicht verwendet werden, wenn die Verbindung zum Server oder zum Web-Client dadurch geschlossen wurde, dass einer der Verbindungsteilnehmer in einer Anforderung bzw. Antwort einen Header 'Connection: close' gesendet hat.

Anweisungen für die korrekte Verwendung dieses Befehls enthalten die folgenden Quellen:

- Lesen Sie zum Schreiben von Headern für eine HTTP-Antwort, die durch CICS als HTTP-Server gesendet wird, den Abschnitt [HTTP-Header für eine Antwort schreiben](#).
- Lesen Sie zum Schreiben von Headern für eine HTTP-Anforderung, die durch CICS als HTTP-Client gesendet wird, den Abschnitt [HTTP-Anforderungen über CICS als HTTP-Client absetzen](#).
- Lesen Sie bei Verwendung von Chunked Transfer Coding zum Senden einer HTTP-Anforderung oder -Antwort den Abschnitt [HTTP-Anforderung oder -Antwort mit Chunked Transfer Coding senden](#). Dort ist das korrekte Verfahren für das Schreiben von abschließenden Headern für eine in Blöcke aufgeteilte Nachricht beschrieben.

Optionen

HTTPHEADER(datenbereich)

Gibt den Namen des HTTP-Headers an, der zur Anforderung bzw. Antwort hinzugefügt werden soll. Der Name (bestehend aus einer Textzeichenfolge) sollte den Standards der von Ihnen verwendeten HTTP-Spezifikation entsprechen.

NAMELENGTH(datenwert)

Gibt als Vollwort-Binärwert die Länge des Namens für den HTTP-Header an.

SESTOKEN(datenwert)

Wenn CICS als HTTP-Client eingesetzt wird, ist diese Option erforderlich. Sie gibt den 8 Byte umfassenden Binärwert für das Sitzungstoken an, das eine Verbindung zwischen CICS und einem Server eindeutig kennzeichnet. Dieser Wert wird durch einen Befehl **WEB OPEN** für CICS als HTTP-Client zurückgegeben. Die Verwendung des Sitzungstokens ist im Abschnitt [Sitzungstokens](#) erläutert.

VALUE(datenbereich)

Gibt den Wert des benannten HTTP-Headers an. Der Wert (bestehend aus einer Textzeichenfolge) sollte den Standards der von Ihnen verwendeten HTTP-Spezifikation entsprechen.

VALUELENGTH(datenwert)

Gibt als Vollwort-Binärwert die Länge des Wertes für den HTTP-Header an.

Bedingungen

16 INVREQ

Die RESP2-Werte lauten wie folgt:

1

Der Befehl wurde in einer Anwendung ausgegeben, die nicht zur CICS-Webunterstützung gehört.

6

Der Client hat in der Anforderung 'TE: trailers' nicht gesendet, weshalb keine abschließenden Header verwendet werden können.

19

Der Header ist nicht zulässig. Manche Anforderungsheader können nur durch CICS generiert werden.

44

Der Header ist nicht als abschließender Header (Trailer) zulässig.

69

Das Chunked Transfer Coding wird nicht unterstützt.

70

Der Header 'Trailer' wurde nicht erstellt, weshalb keine abschließenden Header geschrieben werden können.

71

Beim Chunked Transfer Coding trat ein Fehler auf.

74

Der vorherige Sendevorgang ist fehlgeschlagen.

78

Es ist zu spät, um abschließende Header für diese Nachricht zu schreiben.

22 LENGERR

Die RESP2-Werte lauten wie folgt:

35

Die in der Option NAMELENGTH angegebene Länge ist nicht größer als Null.

55

Die in der Option VALUELENGTH angegebene Länge ist nicht größer als Null oder größer als 32000.

19 NOTOPEN

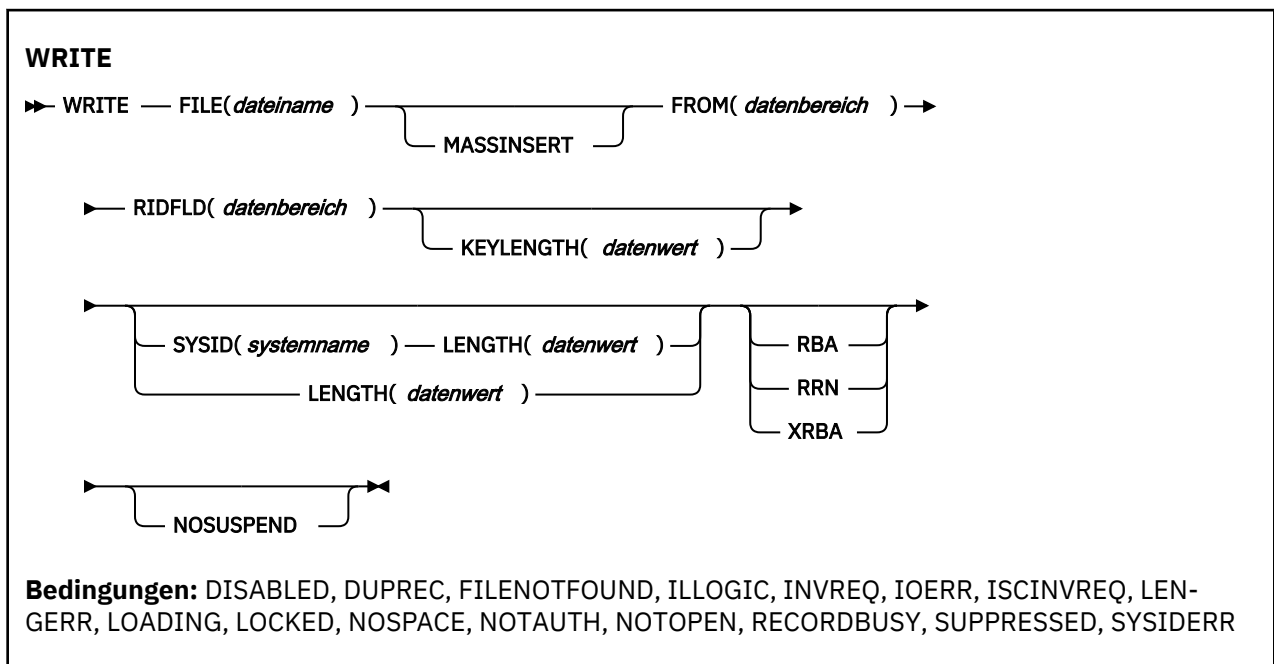
Die RESP2-Werte lauten wie folgt:

27

Das Sitzungstoken ist ungültig.

WRITE

Schreibt einen Datensatz.



Dieser Befehl ist threadsicher, wenn für die Datei, auf die er sich bezieht, Folgendes zutrifft:

- Sie ist als fern definiert und der Befehl wird über eine IPIC-Verbindung durch Funktionsverlagerung an eine ferne CICS-Region übertragen.
- Sie ist als lokale VSAM-Datei, als VSAM-RLS-Datei oder als Coupling-Facility-Datentabelle definiert.

Dieser Befehl ist nicht threadsicher, wenn für die Datei, auf die er sich bezieht, Folgendes zutrifft:

- Sie ist als fern definiert und der Befehl wird über eine Nicht-IPIC-Verbindung durch Funktionsverlagerung übertragen.
- Sie ist als gemeinsam genutzte Datentabelle oder als BDAM-Datei definiert.

Beschreibung

Der Befehl WRITE schreibt einen neuen Datensatz in eine Datei auf einem lokalen oder fernen System.

Wenn dieser Befehl eingesetzt wird, um einen Datensatz in eine von CICS verwalteten Datentabelle zu schreiben, wird die Aktualisierung sowohl an der schlüsseladressierten VSAM-Quellendatei als auch an der speicherinternen Datentabelle vorgenommen, sofern der Benutzerexit XDTAD den Datensatz nicht aus der Tabelle zurückweist. Die Details des Befehls für eine von CICS verwaltete Tabelle sind dieselben wie für eine schlüsseladressierte VSAM-Datei.

Wenn dieser Befehl eingesetzt wird, um einen Datensatz in eine benutzerverwaltete Datentabelle zu schreiben, wird die Aktualisierung an der speicherinternen Datentabelle vorgenommen (sofern sie nicht durch den Benutzerexit XDTAD zurückgewiesen wird).

Wenn dieser Befehl eingesetzt wird, um einen Datensatz in eine Coupling-Facility-Datentabelle zu schreiben, wird ausschließlich die Datentabelle in der Coupling-Facility aktualisiert (sofern nicht durch den Benutzerexit XDTAD zurückgewiesen).

Falls ein Schreibvorgang für eine benutzerverwaltete oder Coupling-Facility-Datentabelle aufgrund fehlenden Speicherbereichs erfolglos ist, wird an den Befehl die Antwort NOSPACE zurückgegeben. CICS gibt außerdem die Nachricht DFHFC0432 aus, um Systemadministratoren über die Situation zu benachrichtigen.

Bei einer Datei im VSAM-Format mit Zugangsfolge oder im erweiterten VSAM-Format mit Zugangsfolge und erweiterter Adressierung wird der Datensatz immer am Ende der Datei hinzugefügt. CICS verwendet bei der Berechnung der relativen Byteadresse (RBA) für den neuen Datensatz nicht das in RIDFLD angegebene Kennungsfeld bzw. die erweiterte relative Byteadresse (XRBA) für eine Datei mit Zugangsfolge und erweiterter Adressierung. Die neue relative Byteadresse bzw. erweiterte relative Byteadresse wird jedoch in dem Satz-ID-Feld, das in der Option RIDFLD angegeben ist, an die Anwendung zurückgegeben.

Bei einer schlüsseladressierten VSAM-Datei wird der Datensatz an der Position hinzugefügt, die durch den zugehörigen Schlüssel angegeben ist; hierbei kann es sich um eine beliebige Position in der Datei handeln. Bei VSAM-Dateien müssen der Schlüssel im Datensatz und der Schlüssel im Kennungsfeld RIDFLD identisch sein.

Bei einer schlüsseladressierten VSAM-Datei (KSDS) oder einer VSAM-Datei in Zugangsfolge (ESDS) können die Datensätze entweder eine feste oder eine variable Länge aufweisen. Operationen MASSINSERT müssen mit aufsteigenden Schlüsseln fortfahren und durch einen Befehl UNLOCK beendet werden, bevor eine andere Anforderung an denselben Datensatz gesendet wird.

Optionen

FILE(dateiname)

Gibt den Namen der Datei an, auf die zugegriffen werden soll.

Bei Angabe eines Wertes für SYSID wird davon ausgegangen, dass sich die Datei, auf die sich dieser Dateiname bezieht, auf einem fernen System befindet, wobei es ohne Belang ist, ob der Name für CICS definiert ist. Ansonsten wird die Ressourcendefinition verwendet, um zu ermitteln, ob sich die Datei (Dataset) auf einem lokalen oder fernen System befindet.

FROM(datenbereich)

Gibt den Datensatz an, der in die durch diesen Namen angegebene Datei geschrieben werden soll.

KEYLENGTH(datenwert)

Gibt die Länge (Halbwort-Binärzahl) des Schlüssels an, der in der Option RIDFLD angegeben ist. Ausgenommen hiervon ist die Angabe der relativen Byteadresse (RBA) oder der relativen Satznummer (RRN); in diesem Fall ist die Option KEYLENGTH nicht gültig. Die Option KEYLENGTH muss codiert werden, wenn Sie ebenfalls die Option SYSID verwenden (es sei denn, Sie verwenden ebenfalls RBA oder RRN). Falls die angegebene Länge von der für die Datei definierten Länge abweicht, tritt die Bedingung INVREQ auf. Die Klausel KEYLENGTH ist erforderlich, wenn der Befehl WRITE FILE per Funktionsverlagerung übertragen wird; andernfalls tritt möglicherweise eine Bedingung INVREQ mit RESP2=23 auf.

LENGTH(datenwert)

Gibt als Halbwort-Binärwert die Länge des Datenbereichs an, aus dem der Datensatz geschrieben werden soll.

Bei Angabe von SYSID muss diese Option angegeben werden.

Falls sich die Datei auf einem fernen System befindet und die Option SYSID *nicht* angegeben ist, muss der Parameter LENGTH hier nicht festgelegt werden, aber in der FILE-Ressourcendefinition festgelegt sein.

Falls sich die Datei auf einem lokalen System befindet, muss die Option LENGTH für Datensätze mit variabler Länge angegeben werden, ist für Datensätze mit fester Länge jedoch optional. Die Angabe der Länge für Datensätze mit fester Länge ist jedoch ratsam, weil CICS hierdurch überprüft, dass die Länge des zu schreibenden Datensatzes die für die Datei geltende Definition nicht überschreitet.

Falls bei einem Befehl WRITE für eine Datei mit Datensätzen fester Länge eine falsche Länge angegeben ist, wird ein Datensatz mit der festen Länge geschrieben und die Bedingung LENGERR tritt auf. Wenn Sie zu viele Daten angeben, wird der Datensatz abgeschnitten. Geben Sie zu wenige Daten an, wird der Datensatz mit binären Nullen aufgefüllt.

MASSINSERT

(VSAM) Gibt an, dass der Befehl WRITE Teil einer Masseneinfügeoperation ist, also einer Reihe von Befehlen WRITE, bei denen jeweils die Option MASSINSERT angegeben ist.

Informationen zur Verwendung der Option MASSINSERT bei Dateien, die im RLS-Zugriffsmodus geöffnet sind, enthält der Abschnitt Sperrung der RLS-Datensatzebene.

Für benutzerverwaltete oder Coupling-Facility-Datentabellen kann die Option MASSINSERT nicht verwendet werden.

Anmerkung: Verwenden Sie die Option MASSINSERT nicht bei einem gemeinsamen Zugriff mit erforderlichen Programmen, die Datensätze in leere VSAM-Dateien schreiben, weil der Wechsel des Tasksteuerblocks vom offenen Tasksteuerblock zum QR-Tasksteuerblock und zurück für jede Schreiboperation stattfindet, während sich die Dateien im Lademodus befinden. Zur Vermeidung des Tasksteuerblockwechsels sollten Sie einen einzelnen Befehl WRITE ohne Masseneinfügung verwenden. Weitere Informationen finden Sie unter Leere VSAM-Dateien laden.

NOSUSPEND (nur RLS)

Die Anforderung wartet nicht, falls VSAM eine aktive Sperre für den Datensatz hält (hierzu gehören auch Datensätze, die infolge einer Bedingung DEADLOCK gesperrt sind).

Eine Task könnte beim Ausgeben einer Anforderung WRITE warten, falls der Schlüssel für einen Datensatz bestimmt ist, der gegenwärtig durch eine andere Task geändert, erstellt oder gelöscht wird, weil VSAM immer zuerst die Sperre anfordert.

Anmerkung: Anforderungen, die die Option NOSUSPEND angeben, warten mindestens 1 Sekunde, bevor CICS die Antwort RECORDBUSY zurückgibt.

RBA

(Nur VSAM-Dateien mit Zugangsfolge) Gibt an, dass das in der Option RIDFLD angegebene Feld für die Satzkennung eine relative Byteadresse enthält. Verwenden Sie diese Option nur beim Schreiben von Daten in eine Basis mit Zugangsfolge.

RIDFLD(datenbereich)

Gibt das Feld für die Satzkennung an. Der Inhalt kann ein Schlüssel, eine relative Byteadresse oder eine relative Satznummer (für VSAM-Dateien) bzw. ein Blockverweis, ein physischer Schlüssel und ein Entblockungsargument (für BDAM-Dateien) sein. Bei einer relativen Byteadresse oder einer relativen Satznummer muss dieses Feld das Format einer Vollwort-Binärzahl besitzen. Falls RBA oder XRBA angegeben ist, ist RIDFLD ein Ausgabefeld, das die relative Byteadresse (größer-gleich Null) des Datensatzes enthält, wenn der Befehl erfolgreich ausgeführt wird. Dieser RBA-Wert wird durch CICS automatisch berechnet. Bei Angabe von RRN enthält RIDFLD die relative Satznummer (größer-gleich 1) des zu schreibenden Datensatzes.

Weitere Informationen zum Definieren des Feldes für die Satzkennung finden Sie unter BDAM-Datensätze angeben und VSAM-Datensätze angeben.

Wenn Datensätze zu einer Datei mit Schlüssel hinzugefügt werden, muss das Feld den vollständigen Schlüssel enthalten.

RRN

(Schlüsseladressierte VSAM-Dateien) Gibt an, dass das in der Option RIDFLD angegebene Feld für die Satznummer eine relative Satznummer enthält.

SYSID(systemname)

Gibt den Namen des Systems an, an das die Anforderung gerichtet ist.

Falls Sie die Option SYSID angeben und die Optionen RBA, XRBA sowie RRN weglassen, müssen Sie ebenfalls die Optionen LENGTH und KEYLENGTH angeben, weil diese nicht in der Ressourcendefinition zu finden sind.

Die Option LENGTH muss entweder explizit angegeben sein oder unter Verwendung des Längenattributverweises in Assemblersprache bzw. STG und CSTG in PL/I als Standardwert aus der Option FROM übernommen werden können. In der Programmiersprache C muss LENGTH explizit angegeben sein.

XRBA

Gibt an, dass das in der Option RIDFLD angegebene Feld für die Satznummer eine erweiterte relative Byteadresse enthält. Verwenden Sie diese Option, wenn Sie einen Datensatz in eine Datei mit Zugangsfolge und erweiterter Adressierung schreiben.

Bedingungen

84 DISABLED

RESP2-Werte:

50

Die Datei war anfänglich als inaktiviert definiert und wurde seither nicht aktiviert oder die Datei wurde durch einen Befehl SET FILE oder CEMT SET FILE inaktiviert.

Standardaktion: Die Task wird abnormal beendet.

14 DUPREC

RESP2-Werte:

150

Es wurde versucht, einen Datensatz zu einer Datei hinzuzufügen, indem auf eine Datei oder einen Pfad zu einer Datei (mit dem Attribut UNIQUEKEY) verwiesen wurde, in der/dem derselbe Schlüssel bereits vorhanden ist.

Diese Bedingung tritt auch für eine Coupling-Facility-Datentabelle auf, die das Konfliktmodell verwendet, selbst dann, wenn eine andere Task den Datensatz mit demselben Schlüssel für eine Aktualisierung eingelesen hat. (Bei einer Coupling-Facility-Datentabelle, die das Sperrenmodell verwendet, und bei allen anderen Arten von Dateien wird eine Sperre eingerichtet, falls eine andere Task den Datensatz zur Aktualisierung eingelesen hat, und die Anforderung WRITE wartet auf die Freigabe der Sperre, statt sofort eine Antwort DUPREC zurückzugeben.)

Standardaktion: Die Task wird abnormal beendet.

12 FILENOTFOUND

RESP2-Werte:

1

Ein in der Option FILE referenzierter Dateiname ist nicht für CICS definiert.

Standardaktion: Die Task wird abnormal beendet.

21 ILLOGIC

RESP2-Werte (VSAM):

110

Es ist ein VSAM-Fehler aufgetreten, der zu keiner der anderen CICS-Antwortkategorien gehört.

Weitere Informationen sind bei EIBRCODE im EXEC-Schnittstellenblock verfügbar; Details finden Sie unter [EIB-Felder](#).

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

RESP2-Werte:

20

Gemäß der Ressourcendefinition sind Hinzufügeoperationen nicht zulässig.

23

Beim Schreiben von Datensätzen, die eingebettete Schlüssel enthielten, stimmten der Schlüssel im Satzbereich (Option FROM) und in RIDFLD nicht überein.

26

Die Option KEYLENGTH wurde angegeben und die angegebene Schlüssellänge ist nicht mit der Länge identisch, die für die Datei definiert ist, auf die sich diese Dateiangabe bezieht.

38

Ein Befehl WRITE mit der Option MASSINSERT wurde für eine BDAM-Datei ausgegeben.

40

Für einen BDAM-Schlüssel trat ein Konvertierungsfehler auf.

44

Der Befehl WRITE hat nicht das Format eines Befehls WRITE für eine benutzerverwaltete oder Coupling-Facility-Datentabelle (beispielsweise ist die Option MASSINSERT oder RBA angegeben).

51

Ein Befehl WRITE, der das Schlüsselwort RBA angab, wurde für eine schlüsseladressierte Datei ausgegeben, auf die im RLS-Modus zugegriffen wird. Der Zugriff auf schlüsseladressierte Dateien über die relative Byteadresse wird beim RLS-Modus nicht unterstützt.

55

Die Leseintegrität NOSUSPEND ist nicht zulässig, weil es sich nicht um eine VSAM-Datei handelt, auf die im RLS-Modus zugegriffen wird.

56

Der Versuch, eine wiederherstellbare Coupling-Facility-Datentabelle zu aktualisieren, ist fehlgeschlagen, weil die aktuelle Arbeitseinheit bereits 1024 wiederherstellbare Coupling-Facility-Datentabellen aktualisiert hat. Innerhalb einer einzigen Arbeitseinheit können nicht mehr als 1024 wiederherstellbare Coupling-Facility-Datentabellen aktualisiert werden.

57

Operationen WRITE sind nicht zulässig, weil die zugehörige VSAM-Datei für den Status AVAILABILITY den Wert RREPL aufweist, der solche Operationen ausschließlich durch Replikationsprogramme zulässt.

59

Die Option XRBA war angegeben, bei der Datei handelt es sich jedoch nicht um eine Datei mit Zugangsfolge und erweiterter Adressierung.

Standardaktion: Die Task wird abnormal beendet.

17 IOERR

RESP2-Werte:

120

Während der Dateisteuerungsoperation ist ein E/A-Fehler aufgetreten. Ein E/A-Fehler ist ein nicht erwartetes Ereignis, das nicht durch eine CICS-Bedingung abgedeckt ist. Weitere Informationen sind im EXEC-Schnittstellenblock verfügbar; Details finden Sie unter [EIB-Felder](#).

Bei VSAM-Dateien gibt die Bedingung IOERR für gewöhnlich einen Hardwarefehler an.

Bei BDAM-Dateien könnte die Bedingung IOERR bedeuten, dass Sie versucht haben, den Schreibvorgang für eine BDAM-Trackadresse auszuführen, die für die Datei nicht definiert ist.

Bei einer Coupling-Facility-Datentabelle gibt eine Bedingung IOERR an, dass eine falsche Antwort von einem Coupling-Facility-Zugriff zurückgegeben wurde.

Standardaktion: Die Task wird abnormal beendet.

54 ISCINVREQ

RESP2-Werte:

70

Das ferne System gibt einen Fehler an, der keiner bekannten Bedingung entspricht.

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

RESP2-Werte:

12

Die angegebene Länge für die Schreiboperation überschreitet die maximale Datensatzgröße; der Datensatz wird abgeschnitten.

10

Die Option LENGTH war nicht angegeben. Die Option LENGTH muss bei einem Befehl WRITE für eine Datei mit Datensätzen variabler Länge oder für eine BDAM-Datei mit Datensätzen nicht definierten Formats angegeben sein.

14

Bei einem Befehl WRITE für eine Datei mit Datensätzen fester Länge wurde eine falsche Länge angegeben. Es wurde ein Datensatz mit der festen Länge geschrieben. Falls Sie zu viele Daten angegeben haben, wurde der Datensatz abgeschnitten. Falls Sie zu wenig Daten angegeben haben, wurde der Datensatz mit binären Nullen aufgefüllt.

94 LOADING

RESP2-Werte:

104

Die Anforderung kann nicht erfüllt werden, weil sie für eine Datentabelle ausgegeben wurde, die noch geladen wird. Die Bedingung kann aus einem der folgenden Gründe auftreten:

- Der Befehl WRITE hat einen Datensatzschlüssel angegeben, der außerhalb des gültigen Bereichs der Datensätze liegt, die bislang in eine Coupling-Facility-Datentabelle geladen wurden. Datensätze können während des Ladevorgangs einer Coupling-Facility-Datentabelle nur dann hinzugefügt werden, wenn der angegebene Schlüssel im Bereich derjenigen Datensätze zu finden ist, die bereits geladen worden sind.

Die Antwort LOADING kann auch für eine Coupling-Facility-Datentabelle zurückgegeben werden, deren Ladevorgang fehlgeschlagen ist. Weitere Informationen zu den Folgen, die sich beim Fehlschlagen des Ladevorgangs für eine Coupling-Facility-Datentabelle ergeben, können Sie der Beschreibung für den globalen Benutzerexit XD TLC unter [Exits zur Datentabellenverwaltung \(XDTRD, XDTAD und XD TLC\)](#) entnehmen.

- Ein Befehl WRITE wurde für eine benutzerverwaltete Datentabelle ausgegeben, die gegenwärtig geladen wird. Die Änderung einer benutzerverwalteten Datentabelle während des Ladevorgangs ist nicht möglich.

Falls Ihr Anwendungsprogramm die Bedingung LOADING ständig oder zu häufig feststellt, vergewissern Sie sich, dass dies nicht durch kollidierende FILE-Definitionen verursacht wird, die dieselbe Datei referenzieren.

Standardaktion: Die Task wird abnormal beendet.

100 LOCKED

RESP2-Werte:

106

Es wurde versucht, einen Datensatz zu schreiben, aber für den Schlüssel dieses Datensatzes besteht eine beibehaltene Sperre.

Standardaktion: Die Task wird mit dem Code AEX8 abnormal beendet.

18 NOSPACE

RESP2-Werte:

100

Auf der Direktzugriffseinheit ist kein Speicherbereich für das Hinzufügen von Datensätzen zu einer Datei verfügbar.

102

Die angegebene maximale Anzahl für Tabelleneinträge für die benutzerverwaltete oder Coupling-Facility-Datentabelle wurde bereits erreicht.

Diese Bedingung kann ebenfalls für eine wiederherstellbare Coupling-Facility-Datentabelle auftreten, wenn die Tabelle scheinbar weniger als die zulässige maximale Anzahl von Datensätzen enthält, falls es noch nicht festgeschriebene Aktualisierungen gibt.

103

CICS kann nicht genügend Speicher im CICS-Adressraum abrufen, um für den geschriebenen Datensatz einen Eintrag in der M-Tabelle zu erstellen.

108

Im Pool für Coupling-Facility-Datentabellen ist nicht genügend Speicherbereich vorhanden, um den Datensatz zu speichern.

Standardaktion: Die Task wird abnormal beendet.

70 NOTAUTH

RESP2-Werte:

101

Eine Ressourcensicherheitsprüfung ist für FILE(dateiname) fehlgeschlagen.

Standardaktion: Die Task wird abnormal beendet.

19 NOTOPEN

RESP2-Werte:

60

Die Bedingung NOTOPEN (RESP2 60) wird aus einem der folgenden Gründe zurückgegeben:

- Die angeforderte Datei ist geschlossen (CLOSED) und nicht aktiviert (UNENABLED). Der Status CLOSED, UNENABLED wird erreicht, nachdem eine Anforderung CLOSE für eine Datei mit dem Status OPEN ENABLED empfangen wurde und die Datei nicht mehr verwendet wird. Sie können CLOSED, UNENABLED auch als Anfangsstatus festlegen, indem Sie in der FILE-Ressourcendefinition STATUS(UNENABLED) und OPENTIME(FIRSTREF) angeben. (Bei BDAM-Dateien verwenden Sie den Parameter FILSTAT des Makros DFHFCT (TYPE=FILE)).
- Die angeforderte Datei besitzt den Status OPEN (geöffnet) und wird durch andere Transaktionen verwendet, aber für die Datei wurde eine Anforderung CLOSE empfangen.
- Eine Anforderung WRITE wurde für eine Datei ausgegeben, die sich aufgrund eines Befehls SET DSNAME QUIESCED oder IMMQUIESCED im Quiescemodus befindet oder gerade in diesen Modus versetzt wird.
- Die angeforderte Datei hat den Status CLOSED (geschlossen) und ENABLED (aktiviert), weshalb CICS versucht hat, die Datei im Rahmen der Anforderungsausführung zu öffnen. Das Öffnen der Datei ist aus einem nicht näher bezeichneten Grund fehlgeschlagen. Sie sollten in der Konsole nach Nachrichten suchen, die erläutern, warum die Datei nicht erfolgreich geöffnet werden konnte.

Diese Bedingung tritt nicht auf, wenn die Anforderung für eine geschlossene, nicht aktivierte Datei (CLOSED, DISABLED) abgesetzt wird. In diesem Fall tritt die Bedingung DISABLED auf.

Standardaktion: Die Task wird abnormal beendet.

101 RECORDBUSY

RESP2-Werte:

107

Die Leseintegrität NOSUSPEND ist in der Anforderung angegeben, aber VSAM hält eine aktive Sperre für den Datensatz, was einen Wartestatus für die Anforderung zur Folge hätte. Weitere Informationen enthält der Abschnitt Beibehaltene und aktive Sperren.

Standardaktion: Die Task wird mit dem Code AEX9 abnormal beendet.

72 SUPPRESSED

RESP2-Werte:

105

Ein am Exitpunkt XDTAD aufgerufenes Benutzerexitprogramm hat festgelegt, dass der Datensatz nicht zur benutzerverwalteten oder Coupling-Facility-Datentabelle hinzugefügt wird.

Standardaktion: Die Task wird abnormal beendet.

53 SYSIDERR

RESP2-Werte:

130

Die Option SYSID gibt einen Namen an, bei dem es sich weder um die lokale CICS-Region noch um ein fernes System handelt, das für CICS durch eine CONNECTION-Definition definiert ist. Die Bedingung SYSIDERR tritt außerdem auf, wenn die Verbindung zu dem fernen System geschlossen ist.

131

Für eine Coupling-Facility-Datentabelle ist die Verbindung zum Server für Coupling-Facility-Datentabellen fehlgeschlagen. Dies könnte dadurch verursacht worden sein, dass der Server selbst fehlgeschlagen ist, oder dass der Server zwar verfügbar ist, CICS jedoch keine Verbindung zu ihm herstellen konnte.

132

Der Befehl WRITE wurde für eine Coupling-Facility-Datentabelle ausgegeben, die nicht mehr vorhanden ist. Vermutlich liegt dies an einem Fehler der Coupling-Facility, durch den der Server für Coupling-Facility-Datentabellen ebenfalls fehlschlägt. Informationen zum erneuten Starten eines Servers für Coupling-Facility-Datentabellen und zum erneuten Laden einer Tabelle finden Sie unter [Server für Coupling-Facility-Datentabellen einrichten und ausführen](#).

Standardaktion: Die Task wird abnormal beendet.

Beibehaltene und aktive Sperren: RECORDBUSY bezieht sich auf aktive Sperren, LOCKED bezieht sich auf beibehaltene Sperren:

- Anforderungen READNEXT für Datensätze mit *beibehaltenen* Sperren werden immer mit einer Antwort LOCKED zurückgewiesen.
- Anforderungen READNEXT für Datensätze mit *aktiven* Sperren warten auf die Freigabe der Sperre. Ausgenommen sind Fälle, bei denen das Schlüsselwort NOSUSPEND angegeben ist; hier gibt CICS die Antwort RECORDBUSY zurück.

Beispiele

Das folgende Beispiel zeigt einen einfachen Befehl WRITE:

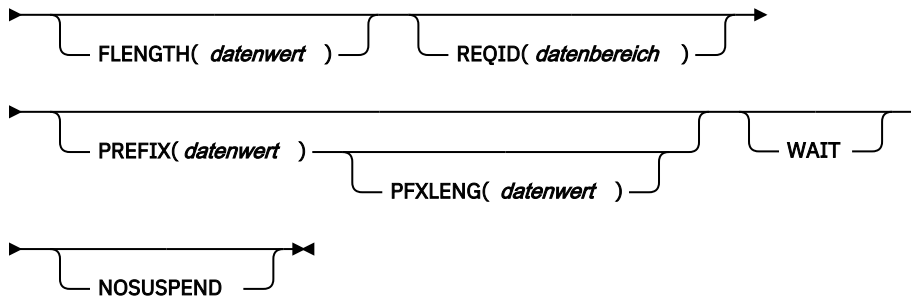
```
EXEC CICS WRITE  
      FROM(RECORD)  
      LENGTH(DATLEN)  
      FILE('MASTER')  
      RIDFLD(KEYFLD)
```

WRITE JOURNALNAME

Erstellt einen Journalsatz.

WRITE JOURNALNAME

➔ WRITE JOURNALNAME(*datenwert*) — JTYPEID(*datenwert*) — FROM(*datenbereich*) —➔



Bedingungen: INVREQ, IOERR, JIDERR, LENGERR, NOJBUFSP, NOTAUTH, NOTOPEN

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl WRITE JOURNALNAME schreibt einen Journalsatz aus dem angegebenen Datenbereich in den Protokolldatenstrom der Systemprotokollfunktion, die dem CICS-Journalnamen oder SMF entspricht. Die Anforderung kann sich auf synchrone oder asynchrone Ausgabe beziehen; Definitionen dieser Begriffe sowie Informationen zur Synchronisation von Journalausgabe finden Sie im Abschnitt [Synchronisation der Journalausgabe](#).

Optionen

FLENGTH(*datenwert*)

Gibt als Vollwort-Binärwert die Länge der Benutzerdaten in Byte an, die im Journalsatz erstellt werden sollen.

Die maximale Gesamtlänge eines Journalsatzes ist von einer Reihe Faktoren abhängig:

- Falls das Journal SMF verwendet, beträgt der Grenzwert 32 KB minus 400 Byte.
- Der Grenzwert für Journale, die Protokolldatenströmen zugeordnet sind, ist der im Attribut MAX-BUFSIZE für die verwendete Struktur ausgedrückte Wert minus 400 Byte. Dieser Wert muss die Benutzerdaten, die Präfixdaten und den 2 Byte umfassenden Wert für JTYPEID enthalten.

Anmerkung: Daten, die länger als 32 K sind, können durch Offline-Jobs unter Verwendung der Schnittstelle SUBSYS=LOGR nicht gelesen werden.

FROM(*datenbereich*)

Gibt die Benutzerdaten an, die im Journalsatz erstellt werden sollen.

JOURNALNAME(*datenwert*)

Gibt einen 1 bis 8 Zeichen langen Journalnamen an. Gültige Zeichen für einen Journalnamen sind die Großbuchstaben A bis Z, die Ziffern 0 bis 9 sowie die Sonderzeichen \$ @ und #.

Beim ersten Verweis auf diesen Journalnamen muss CICS in der Lage sein, den Journalnamen zu einem korrespondierenden Protokolldatenstrom für die MVS-Systemprotokollfunktion oder zur MVS-SMF-Datei zuzuordnen. Hierzu durchsucht CICS die installierten JOURNALMODEL-Definitionen nach einem übereinstimmenden Journalnamen in einem Journalmodell. CICS sucht entweder nach einer bestimmten Übereinstimmung oder nach einer generischen Übereinstimmung. Falls kein übereinstimmender Eintrag gefunden wird, versucht CICS, einen Standardprotokolldatenstromnamen zu verwenden.

Um den Eintrag in das CICS-Systemprotokoll zu schreiben, geben Sie DFHLOG als Journalnamen an.

Anmerkung: Das CICS-Systemprotokoll sollte nur für kurzlebige Daten verwendet werden, die für Wiederherstellungszwecke erforderlich sind. Benutzerdatensätze für Dinge wie Auditprotokolle sollten nicht in dieses Protokoll geschrieben werden.

Um Einträge in Journale zu schreiben, die mit der Konvention für die Journalnummerierung definiert sind (z. B. die in FILE-Ressourcendefinitionen definierten automatischen Journale), geben Sie als Name DFHJnn an, wobei 'nn' für die Journalnummer zwischen 1 und 99 steht.

Ein Eintrag kann nur dann in ein Protokoll für die aktualisierende Wiederherstellung geschrieben werden, das CICS nur unter seinem 26 Zeichen langen Namen des Protokolldatenstroms kennt (der direkt aus dem VSAM-ICF-Katalog abgeleitet wird), wenn der Schreibvorgang in einem Journal erfolgt, dessen übereinstimmender Wert für JOURNALMODEL demselben Namen des Protokolldatenstroms zugeordnet ist.

Die Angabe von DFHJ01 bei diesem Befehl bezieht sich auf ein Benutzerjournal und *nicht* auf das Systemprotokoll.

JTYPEID(datenwert)

Gibt eine zweistellige Kennung an, die in den Journalsatz aufgenommen werden soll, um seinen Ursprung anzugeben.

NOSUSPEND

Gibt an, dass das Anwendungsprogramm bei der Bedingung NOJBUFSP nicht ausgesetzt werden soll. Der Benutzerdatensatz wird ignoriert.

PFXLENG(datenwert)

Gibt als Halbwort-Binärwert die Länge der Benutzerpräfixdaten in Byte an, die in den Journalsatz einbezogen werden sollen.

Die maximale Gesamtlänge eines Journalsatzes ist von einer Reihe Faktoren abhängig:

- Falls das Journal SMF verwendet, beträgt der Grenzwert 32 KB minus 400 Byte.
- Der Grenzwert für Journale, die Protokolldatenströmen zugeordnet sind, ist der im Attribut MAX-BUFSIZE für die verwendete Struktur ausgedrückte Wert minus 400 Byte. Dieser Wert muss die Benutzerdaten, die Präfixdaten und den 2 Byte umfassenden Wert für JTYPEID enthalten.

Der Mindestwert ist 0. Informationen zu den Grenzwerten für die Größe eines Journalsatzes enthält die Beschreibung der Option FLENGTH.

Anmerkung: Daten, die länger als 32 K sind, können durch Offline-Jobs unter Verwendung der Schnittstelle SUBSYS=LOGR nicht gelesen werden.

PREFIX(datenwert)

Gibt die Benutzerpräfixdaten an, die in den Journalsatz einbezogen werden sollen. In COBOL-Programmen muss ein Datenbereich angegeben werden.

REQID(datenbereich)

Gibt einen Datenbereich an, der den Journalsatz angibt. Der Datenbereich ist eine Vollwort-Binärzahlvariable. CICS legt für die Variable ein Token fest, das zur Synchronisation verwendet werden kann. REQID ist nur für asynchrone Ausgabe gültig (also nur dann, wenn die Option WAIT nicht angegeben ist).

WAIT

Gibt an, dass eine synchrone Journalausgabe erforderlich ist. Die anfordernde Task wartet, bis der Datensatz permanent gespeichert wurde.

Bedingungen

16 INVREQ

Der Bedingung ist für die Verarbeitung durch CICS ungültig.

Standardaktion: Die Task wird abnormal beendet.

17 IOERR

Ein Journalsatz wurde nicht ausgegeben, weil eine Bedingung für einen nicht behebbaren Fehler durch den Protokolldatenstrom für die Systemprotokollfunktion oder durch SMF zurückgegeben wurde.

Standardaktion: Falls es sich bei dem Protokoll um das Systemprotokoll handelt, wird CICS entweder in den Quiesce-Status versetzt oder abgebrochen. Handelt es sich um ein allgemeines Protokoll, wird die Task abnormal beendet.

43 JIDERR

CICS kann aus einem der folgenden Gründe keine Verbindung zu einem Protokolldatenstrom herstellen, der durch den angegebenen Journalnamen referenziert wird:

- Der Protokolldatenstrom ist nicht vorhanden und kann unter Verwendung der Standardmodelldefinitionen nicht dynamisch erstellt werden.
- Der Protokolldatenstrom ist ein reiner DASD-Protokolldatenstrom, mit dem gegenwärtig eine CICS-Region in einem anderen MVS-Image verbunden ist.

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

Die aggregierte Länge des Journalsatzes, die die Benutzerdaten (Daten aus FROM, JTYPE und PREFIX) sowie die CICS-Headerdaten umfasst, ist zu groß für die maximale Blockgröße, die für den Protokolldatenstrom zulässig ist.

Standardaktion: Die Task wird abnormal beendet.

45 NOJBUFSP

Die Journalpuffer sind logisch belegt (der aktuelle Puffer besitzt somit nicht genügend Speicherplatz für diesen Journalsatz und für den alternativen Puffer wird gegenwärtig eine Ein-/Ausgabe verarbeitet).

Standardaktion: CICS setzt die Taskaktivität aus, bis die Journalanforderung erfüllt werden kann. CICS stellt sicher, dass beide Puffer in den Zusatzspeicher ausgelagert und so für neue Datensätze freigegeben werden. (Die Standardaktion kann durch die Option NOSUSPEND außer Kraft gesetzt werden.)

70 NOTAUTH

Eine Ressourcensicherheitsprüfung für den Wert von JOURNALNAME(datenwert) ist fehlgeschlagen.

Standardaktion: Die Task wird abnormal beendet.

19 NOTOPEN

Tritt in einer der folgenden Situationen auf:

- Der Befehl kann nicht ausgeführt werden, weil das angegebene Journal durch den Benutzer explizit inaktiviert wurde.
- Die Anforderung konnte nicht erfüllt werden, weil das angegebene Journal mithilfe eines Journalmodells definiert wurde, das es dem Protokolldatenstrom zuordnet, der als Systemprotokoll für dieses CICS-System verwendet wird. Der Fehler wird bei dem Versuch festgestellt, eine Verbindung zum Protokolldatenstrom herzustellen, und bewirkt, dass eine Definition für JOURNALNAME installiert und auf den Status 'failed' (fehlgeschlagen) gesetzt wird.

Standardaktion: Die Task wird abnormal beendet.

Beispiele

Das folgende Beispiel zeigt einen Befehl, mit dem synchrone Journalausgabe geschrieben und auf den Abschluss der Ausgabeoperation gewartet wird:

Infolge einer Änderung an dem Verfahren, mit dem CICS an die Konsole gesendete Nachrichten verarbeitet, werden Texte mit einer Länge von mehr als 113 Zeichen in zwei Zeilen geteilt. Keine der nachfolgend aufgeführten Variablen kann als variable PL/I-Zeichenfolge definiert sein.

Anmerkung: Falls ACTION (oder einer der unten aufgeführten äquivalenten CVDA-Werte) angegeben wird, wird die Nachricht aufbewahrt, bis der Konsolenbediener sie explizit löscht oder bis CICS beendet wird.

Der Aktionscode ist mit dem Deskriptorcode identisch, der der Nachricht zugeordnet werden soll. Für diesen Parameter kann nur ein einziger der Deskriptorcodes 2, 3 oder 11 angegeben werden.

Falls ACTION nicht angegeben ist, wird der Nachricht kein Deskriptorcode zugeordnet. Die Deskriptorcodes besitzen die folgenden Bedeutungen:

2

Sofortige Aktion

3

Spätere Aktion

11

Kritische spätere Aktion

Die Option CRITICAL ist äquivalent zur Angabe von ACTION(11). Die Option EVENTUAL ist äquivalent zur Angabe von ACTION(3). Die Option IMMEDIATE ist äquivalent zur Angabe von ACTION(2).

Aufbewahrte Nachrichten können durch den Konsolenbediener auf unterschiedliche Weise abgewickelt werden (siehe z/OS MVS System Commands). Informationen zu den Auswirkungen dieses Befehls auf die Darstellung der Anzeige auf der Konsole für den Bediener können Sie bei Ihrem Systemprogrammierer erfragen.

Optionen

ACTION(cvda)

Gibt einen Aktionscode an, der dieser Nachricht zugeordnet ist. Die CVDA-Werte lauten wie folgt:

CRITICAL

Gibt an, dass die Nachricht eine spätere Aktion durch den Bediener erforderlich macht und einen ausreichend kritischen Stellenwert besitzt, um in der Anzeige auf der Konsole zu verbleiben. Die Nachricht verbleibt so lange in der Anzeige, bis sie durch den Bediener gelöscht wird.

EVENTUAL

Gibt an, dass der Bediener die Aktion zu einem geeigneten Zeitpunkt ausführen sollte. Die Nachricht wird ausgelagert, sobald andere Nachrichten die Anzeige ausfüllen, jedoch vom Betriebssystem aufbewahrt, bis sie explizit durch den Bediener gelöscht wird.

IMMEDIATE

Gibt an, dass der Bediener die Aktion sofort ausführen muss. Die Nachricht verbleibt so lange in der Anzeige auf der Konsole, bis sie durch den Bediener gelöscht wird.

MAXLENGTH(datenwert)

Gibt ein Vollwort-Binärfeld an, das die Länge des Antwortbereichs enthält (zwischen 1 und 119 Byte). Bei Angabe der Option REPLY muss die Option MAXLENGTH angegeben werden.

NUMROUTES(datenwert)

Gibt ein Vollwort-Binärfeld an, das die Anzahl der Routing-Codes definiert.

REPLY(datenbereich)

Gibt einen Datenbereich für den Empfang der Bedienerantwort an. Falls Sie diese Option angeben, wird die Anwendung angehalten, bis entweder eine Antwort empfangen wird oder ein mit TIMEOUT angegebener Zeitraum abgelaufen ist.

REPLYLENGTH(datenbereich)

Gibt als Vollwort-Binärwert die tatsächliche Länge der Bedienerantwort an.

ROUTECODES(datenwert)

Gibt ein Feld variabler Länge an. Jeder Code umfasst 1 Byte und enthält eine Binärzahl im Bereich von 1 bis 28. Der Standardwert ist ein einziger und mit 2 angegebener Code. Lediglich bei COBOL-Programmen müssen Sie einen Datenbereich verwenden, der anstelle eines Datenwertes die 1 Byte umfassenden Werte enthält.

TEXT(datenwert)

Gibt einen Datenwert an, der den zu sendenden Text enthält.

Falls der Datenwert mit 'DFHnnnn' oder 'DFHaannnn' beginnt, wird die Nachricht als CICS-Nachricht behandelt und entsprechend neu formatiert.

Bei Verwendung der Umsetzeroption von COBOL2 müssen Sie keinen Datenwert, sondern einen Datenbereich verwenden, in dem der Text enthalten ist, der an den Bediener gesendet werden soll.

TEXTLENGTH(datenwert)

Gibt als Vollwort-Binärwert die Länge des Textes an. Diese Option ist nur bei Programmen in den Programmiersprache C und C++ erforderlich.

- Falls die Option REPLY angegeben ist, liegt die Länge im Bereich von 0 bis 121 Byte.
- Falls die Option REPLY nicht angegeben ist, liegt die Länge im Bereich von 0 bis 690 Byte.

Ist die Länge des Textes größer als 113, formatiert CICS die Nachricht als mehrzeiligen Schreibvorgang an den Bediener (Write To Operator, WTO); jede Zeile umfasst 69 Byte, maximal sind 10 Zeilen möglich.

Die Ausgabe wird so bearbeitet, dass jede Zeile nach Möglichkeit an einem Leerzeichen umgebrochen wird und die nächste Zeile nicht mit einem Leerzeichen beginnt. Falls die Daten nicht innerhalb des Gesamtgrenzwertes von 690 Byte in 10 Zeilen zu 69 Byte formatiert werden können, wird die Ausgabe nicht neu formatiert.

TIMEOUT(datenwert)

Gibt ein Vollwort-Binärfeld an, das die maximale Zeit (in Sekunden) enthält, die CICS auf eine Antwort wartet, bevor die Steuerung an diese Transaktion zurückgegeben wird. Der Wert muss im Bereich von 0 bis 86 400 (= 24 Stunden) liegen. Der Systemstandardwert wird durch den Systeminitialisierungsparameter **OPERTIM** angegeben. Sie können die Option TIMEOUT nur angeben, wenn Sie ebenfalls die Option REPLY angegeben haben.

Bedingungen**31 EXPIRED**

RESP2-Werte:

7

Vor dem Empfang der Bedienerantwort trat eine Zeitlimitüberschreitung auf.

Standardaktion: Die Ausnahmebedingung wird an die Anwendung zurückgegeben.

16 INVREQ

RESP2-Werte:

1

Der Wert für TEXTLENGTH ist ungültig.

2

Der Wert für NUMROUTES ist ungültig.

3

Der Wert für ROUTECODES ist ungültig.

4

Der Wert für MAXLENGTH ist ungültig.

5

Der Wert für TIMEOUT ist ungültig.

6

Der Wert für ACTION ist ungültig.

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

RESP2-Werte:

8

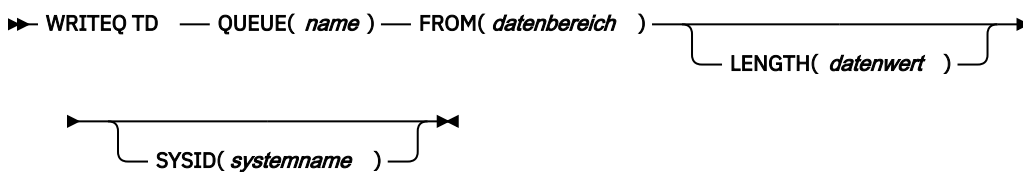
Die Antwort war länger als der mit der Option MAXLENGTH angegebene Wert und wurde abgeschnitten.

Standardaktion: Die Task wird abnormal beendet.

WRITEQ TD

Schreibt Daten in die Warteschlange mit transienten Daten.

WRITEQ TD



Bedingungen: DISABLED, ERROR, INVREQ, IOERR, ISCINVREQ, LENGERR, LOCKED, NOSPACE, NOTAUTH, NOTOPEN, QIDERR, SYSIDERR

Dieser Befehl ist threadsicher, wenn er mit einer Warteschlange in einer lokalen CICS-Region verwendet wird oder wenn er durch Funktionsverlagerung über eine IPIC-Verbindung an eine ferne CICS-Region übertragen wird. Er ist nicht threadsicher, wenn er durch Funktionsverlagerung über einen anderen Typ von Verbindung an eine ferne CICS-Region übertragen wird.

Beschreibung

Der Befehl **WRITEQ TD** schreibt transiente Daten in ein vordefiniertes symbolisches Ziel.

Optionen

FROM(datenbereich)

Gibt die Daten an, die in die Warteschlange mit transienten Daten geschrieben werden sollen.

LENGTH(datenwert)

Gibt als Halbwort-Binärwert die Länge der zu schreibenden Daten an.

QUEUE(name)

Gibt den symbolischen Namen (1 bis 4 alphanumerische Zeichen) der Warteschlange an, in die die Daten geschrieben werden. Die benannte Warteschlange muss für CICS definiert worden sein.

SYSID(systemname)

(Nur ferne Systeme) Gibt den ein bis vier Zeichen langen Namen des Systems an, an das die Anforderung gerichtet ist.

Bei Angabe eines Wertes für SYSID wird davon ausgegangen, dass sich die Warteschlange auf einem fernen System befindet, wobei es ohne Belang ist, ob sie als ferne Warteschlange definiert ist oder nicht. Andernfalls wird mithilfe der Definition für die Warteschlange mit transienten Daten ermittelt, ob sich die Datei auf einem lokalen oder fernen System befindet.

Bedingungen

84 DISABLED

Tritt auf, wenn die Warteschlange inaktiviert wurde.

Standardaktion: Die Task wird abnormal beendet.

1 ERROR

Tritt bei einem Fehler auf, der keine andere Bedingung auslöst.

Standardaktion: Die Task wird abnormal beendet.

16 INVREQ

Tritt auf, falls der Befehl WRITEQ eine partitionsübergreifende Warteschlange benennt, die für die Eingabe geöffnet wurde.

Anmerkung: Für partitionsinterne Warteschlangen kann diese Bedingung nicht ausgelöst werden.

Standardaktion: Die Task wird abnormal beendet.

17 IOERR

Tritt auf, wenn es einen Ein-/Ausgabefehler gibt und der fehlerhafte Datensatz übersprungen wird.

Standardaktion: Die Task wird abnormal beendet.

54 ISCINVREQ

Tritt auf, wenn das ferne System einen Fehler angegeben hat, der keiner bekannten Bedingung entspricht.

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

Tritt in einer der folgenden Situationen auf:

- Der Befehl WRITEQ hat eine partitionsübergreifende Warteschlange benannt und keine Länge angegeben, die mit der Option RECORDSIZE und zugehörigen Formatierungen konsistent ist, die in der TDQUEUE-Ressourcendefinition angegeben sind. Die Prüfung findet statt, nachdem der Exit XTDOU aufgerufen wurde; dieser Exit kann die Länge der Daten ändern, die an die Zugriffsmethode übergeben werden sollen.
- Der Befehl WRITEQ hat eine partitionsinterne Warteschlange benannt und keine Länge angegeben, die mit dem für die partitionsinterne Datei definierten Steuerintervall konsistent ist. Auch hier findet die Prüfung statt, nachdem der Exit XTDOU aufgerufen wurde.

Standardaktion: Die Task wird abnormal beendet.

100 LOCKED

Tritt auf, wenn die Anforderung nicht ausgeführt werden kann, weil die Nutzung der Warteschlange aufgrund des unbestätigten Fehlschlagens einer Arbeitseinheit eingeschränkt wurde. Dies kann bei einer beliebigen Anforderung für eine logisch wiederherstellbare Warteschlange stattfinden, die in der TDQUEUE-Ressourcendefinition mit WAIT(YES) und WAITACTION(REJECT) definiert ist.

Geben Sie WAIT(YES) und WAITACTION(QUEUE) in der TDQUEUE-Ressourcendefinition an, wenn die Transaktion warten soll.

Standardaktion: Die Task wird abnormal beendet.

18 NOSPACE

Tritt auf, falls in der partitionsinternen oder partitionsübergreifenden Warteschlange kein Platz mehr ist oder falls die relative Byteadresse (RBA) für eine partitionsinterne Warteschlange 2 GB überschreiten würde. In diesem Fall sollten keine weiteren Daten in die Warteschlange geschrieben werden, da sie möglicherweise verloren gehen.

Standardaktion: Die Task wird abnormal beendet.

70 NOTAUTH

RESP2-Werte:

101

Eine Ressourcensicherheitsprüfung ist für QUEUE(*name*) fehlgeschlagen.

102

Eine Ersatzsicherheitsprüfung ist fehlgeschlagen. Weitere Informationen finden Sie unter [Sicherheit für die Übergabe eines JCL-Jobs an den internen Leser](#).

Standardaktion: Die Task wird abnormal beendet.

19 NOTOPEN

Tritt auf, wenn die Zieladresse geschlossen ist.

Anmerkung: Für partitionsinterne Warteschlangen kann diese Bedingung nicht ausgelöst werden.

Standardaktion: Die Task wird abnormal beendet.

44 QIDERR

Tritt auf, wenn das symbolische Ziel, das bei einem Befehl für die Steuerung von transienten Daten verwendet werden soll, nicht zu finden ist.

Standardaktion: Die Task wird abnormal beendet.

53 SYSIDERR

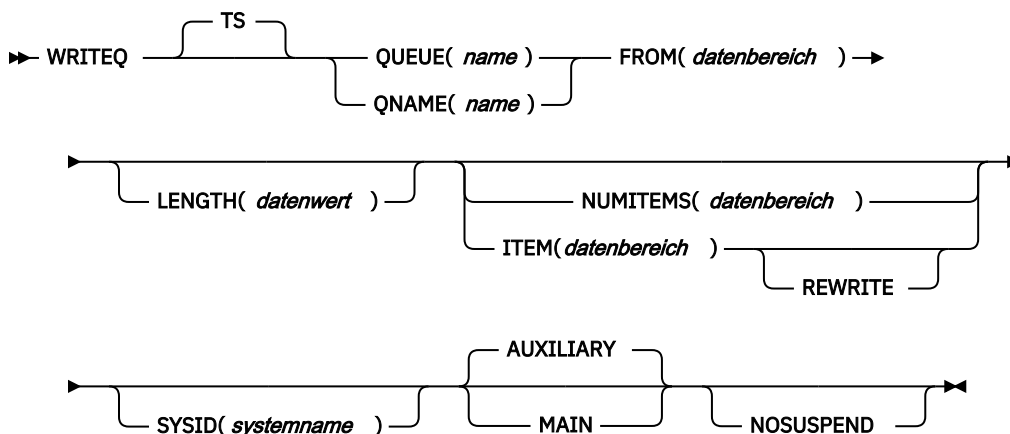
Tritt auf, wenn die Option SYSID einen Namen angibt, bei dem es sich weder um das lokale System noch um ein fernes System handelt (das CICS durch die Definition einer CONNECTION- oder IPCONN-Ressource bekanntgegeben wird). Die Bedingung SYSIDERR tritt außerdem auf, wenn die Verbindung zu dem fernen System geschlossen ist.

Standardaktion: Die Task wird abnormal beendet.

WRITEQ TS

Schreibt Daten in die Warteschlange für temporären Speicher.

WRITEQ TS



Bedingungen: INVREQ, IOERR, ISCINVREQ, ITEMERR, LENGERR, LOCKED, NOSPACE, NOTAUTH, QIDERR, SYSIDERR

Dieser Befehl ist threadsicher, wenn er für eine Warteschlange im Hauptspeicher oder im Zusatzspeicher entweder in einer lokalen CICS-Region verwendet oder mit der Funktionsverlagerung über eine IPIC-Verbindung in eine ferne CICS-Region übertragen wird. Er ist ebenfalls threadsicher, wenn er für eine Warteschlange in einem gemeinsam genutzten temporären Speicherpool in einer z/OS-Coupling-Facility verwendet wird, die durch einen Server für gemeinsame Datennutzung im temporären Speicher (TS-Server) verwaltet wird. Der Befehl ist nicht threadsicher, wenn er durch Funktionsverlagerung über einen anderen Verbindungstyp als IPIC in eine ferne CICS-Region übertragen wird.

Hinweis für dynamisches Transaktionsrouting: Die Verwendung dieses Befehls kann transaktionsübergreifende Affinitäten erzeugen, die die Nutzung des dynamischen Transaktionsroutings beeinträchtigen. Weitere Informationen zu Transaktionsaffinitäten finden Sie unter [Affinität](#).

Beschreibung

Der Befehl **WRITEQ TS** speichert Datensätze in einer Warteschlange für temporären Speicher im Haupt- oder Zusatzspeicher.

Wenn eine Warteschlange als wiederherstellbar definiert ist, darf das Programm keinen Befehl **WRITEQ TS** ausgeben, falls in derselben logischen Arbeitseinheit zuvor ein Befehl **DELETEQ TS** ausgegeben wurde. Nach einem Befehl **DELETEQ TS** darf also ein Befehl **WRITEQ TS** erst nach dem Auftreten eines Synchronisationspunktes ausgegeben werden.

Falls in der Datei des temporären Speichers oder im Hauptspeicher nicht genügend Speicherbereich verfügbar ist, um die Anforderung **WRITEQ TS** zu erfüllen, wird die Task ausgesetzt, bis Speicherbereich verfügbar wird. (Andere Tasks im System geben möglicherweise Speicherbereich frei.) Falls kein Speicherbereich verfügbar ist und Sie die Option **NOSUSPEND** angegeben haben, tritt die Bedingung **NOSPACE** auf und Sie können entscheiden, ob die Transaktion mit einem Abbruch gestoppt oder ob gewartet werden soll.

Optionen

AUXILIARY

Gibt an, dass sich die Warteschlange für temporären Speicher in einer DASD-Einheit im Zusatzspeicher befindet. Dies ist der Standardwert für den ersten Schreibvorgang.

In den folgenden Situationen wird diese Option ignoriert:

- Eine Warteschlange ist vorhanden.
- Im System ist eine TSMODEL-Ressourcendefinition mit einem übereinstimmenden Präfix installiert.
- Die angegebene Datenwarteschlange für temporären Speicher befindet sich in einem gemeinsam genutzten temporären Speicherpool.
- CICS wurde so initialisiert, dass nur Hauptspeicher für temporären Speicher unterstützt wird, es war also der Systeminitialisierungsparameter **TS=(,0)** angegeben.

FROM(datenbereich)

Gibt die Daten an, die in den temporären Speicher geschrieben werden sollen.

ITEM(datenbereich)

Gibt als Halbwort-Binärwert die Elementnummer des logischen Datensatzes an, der in der Warteschlange ersetzt werden soll (bei gleichzeitiger Angabe der Option **REWRITE**).

Die Option **ITEM** kann sowohl ein Eingabefeld als auch ein Ausgabefeld für CICS darstellen. Programmierer müssen daher sicherstellen, dass das Feld **ITEM** nicht im geschützten Speicher definiert ist, wenn ein Befehl **WRITEQ** ausgegeben wird. Falls der Wert für **ITEM** ein Literal war (beispielsweise), würde die Befehlsprüfung (**CMDPROT=YES**) zum Auftreten eines Abbruchs **AEYD** führen.

Anmerkung: In früheren Releases wurde bei Angabe von **ITEM** in einem Befehl **WRITEQ TS** ohne die Option **REWRITE** eine ähnliche Funktion wie **NUMITEMS** ausgeführt. Diese Funktion wurde zur Kompatibilität beibehalten.

LENGTH(datenwert)

Gibt als Halbwort-Binärwert die Länge der zu schreibenden Daten an.

Bei Verwendung der Option **SYSID** müssen Sie diese Option angeben.

Die maximale Länge beträgt 32763. Eine Beschreibung für eine sichere Obergrenze finden Sie unter [LENGTH-Optionen in CICS-Befehlen](#).

MAIN

Gibt an, dass sich die Warteschlange für temporären Speicher im Hauptspeicher befindet.

In den folgenden Situationen wird diese Option ignoriert:

- Eine Warteschlange ist vorhanden.
- Im System ist eine TSMODEL-Ressourcendefinition mit einem übereinstimmenden Präfix installiert.
- Die angegebene Datenwarteschlange für temporären Speicher befindet sich in einem gemeinsam genutzten temporären Speicherpool.

Wenn Sie die Option **MAIN** verwenden, um Daten in eine Warteschlange für temporären Speicher auf einem fernen System zu schreiben, werden die Daten im Hauptspeicher gespeichert, falls der Zugriff

auf das ferne System durch die Funktion für den CICS-Mehrregionenbetrieb (MRO) oder über IPIC-Konnektivität erfolgt. Sind diese Bedingungen nicht erfüllt, werden die Daten im Zusatzspeicher gespeichert.

Falls das System im Mehrregionenbetrieb genutzt wird und die Option MAIN angegeben ist, ist die Warteschlange nicht wiederherstellbar und der Befehl SYNCPOINT ROLLBACK funktioniert nicht.

NOSUSPEND

Gibt an, dass das Anwendungsprogramm nicht ausgesetzt werden soll, wenn in der Datei des temporären Speichers oder im Hauptspeicher nicht genügend Speicherbereich verfügbar ist, um die Anforderung WRITEQ TS zu erfüllen. Stattdessen tritt die Bedingung NOSPACE auf.

Falls jedoch ein Befehl HANDLE CONDITION für NOSPACE aktiv ist, wenn der Befehl ausgeführt wird, setzt diese Bedingung auch die Standardaktion außer Kraft und die Steuerung wird an die Benutzerkennung übergeben, die im Befehl HANDLE CONDITION angegeben ist. Diese Bedingung hat Vorrang vor der Option NOSUSPEND, wird jedoch entweder durch NOHANDLE oder durch RESP natürlich zurückgewiesen.

NUMITEMS(datenbereich)

Gibt ein Feld für eine Halbwort-Binärzahl an, in dem CICS eine Zahl speichert, die angibt, wie viele Elemente sich nach der Ausführung des Befehls WRITEQ TS in der Warteschlange befinden.

Falls der Datensatz eine neue Warteschlange startet, wird die Elementnummer 1 zugeordnet; nachfolgende Elementnummern sind sequenziell. Bei Angabe der Option REWRITE ist die Option NUMITEMS nicht gültig.

QNAME(name)

Als Alternative zu QUEUE gibt die Option QNAME den symbolischen Namen (1 - 16 Zeichen) der Warteschlange an, in die Daten geschrieben werden sollen. Falls der Name kürzer als 16 Zeichen ist, müssen Sie trotzdem ein 16 Zeichen langes Feld verwenden, das bei Bedarf mit Leerzeichen aufgefüllt wird. Falls die Warteschlange für CICS als ferne Warteschlange definiert ist, wird die Anforderung an ein fernes System gesendet. Verwenden Sie als erstes Zeichen im Namen nicht X'FA' bis X'FF' bzw. **, \$\$ oder DF. Diese Zeichen sind für die Verwendung durch CICS reserviert. Der Name kann nicht ausschließlich aus binären Nullen bestehen.

QUEUE(name)

Gibt den symbolischen Namen (1 bis 8 Zeichen) der Warteschlange an, in die Daten geschrieben werden sollen. Falls der Name kürzer als 8 Zeichen ist, müssen Sie trotzdem ein 8 Zeichen langes Feld verwenden, das bei Bedarf mit Leerzeichen aufgefüllt wird. Falls die Warteschlange für CICS als ferne Warteschlange definiert ist, wird die Anforderung an ein fernes System gesendet. Verwenden Sie als erstes Zeichen im Namen nicht X'FA' bis X'FF' bzw. **, \$\$ oder DF. Diese Zeichen sind für die Verwendung durch CICS reserviert. Der Name kann nicht ausschließlich aus binären Nullen bestehen.

REWRITE

Gibt an, dass der vorhandene Datensatz in der Warteschlange durch die bereitgestellten Daten überschrieben werden soll. Falls die Option REWRITE angegeben ist, muss auch die Option ITEM angegeben werden. Wenn die angegebene Warteschlange nicht vorhanden ist, tritt die Bedingung QIDERR auf. Wird in einer vorhandenen Warteschlange nicht das richtige Element gefunden, tritt die Bedingung ITEMERR auf und die Daten werden nicht gespeichert.

SYSID(systemname)

(Nur ferne und gemeinsam genutzte Warteschlangen) Gibt den Systemnamen (1-4 Zeichen) an, der das ferne System oder einen Pool von gemeinsam genutzten Warteschlangen bezeichnet, an das bzw. den die Anforderung übertragen wird. Es ist zu beachten, dass TSMODEL-Ressourcendefinitionen nicht die Angabe eines SYSID-Wertes für eine Warteschlange unterstützen, die sich in einem Pool für die gemeinsame Datennutzung von temporärem Speicher befindet. Verwenden Sie stattdessen die Option QUEUE oder QNAME. Die Verwendung eines expliziten SYSID-Wertes für einen Pool mit gemeinsam genutzten Warteschlangen macht die Unterstützung einer Tabelle für temporären Speicher (TST) erforderlich.

Bedingungen

16 INVREQ

Tritt in einer der folgenden Situationen auf:

- Ein Befehl WRITEQ TS hat einen Warteschlangennamen angegeben, der ausschließlich aus binären Nullen besteht.
- Ein Befehl WRITEQ TS hat eine Warteschlange angegeben, die gesperrt ist und auf die Wiederherstellung der ISC-Sitzung wartet.
- Die Warteschlange wurde durch internen Code von CICS erstellt.

Standardaktion: Die Task wird abnormal beendet.

17 IOERR

RESP2-Werte:

5

Für eine gemeinsam genutzte Warteschlange trat ein nicht wiederherstellbarer Ein-/Ausgabefehler auf.

Standardaktion: Die Task wird abnormal beendet.

54 ISCINVREQ

Tritt auf, wenn das ferne System einen Fehler angegeben hat, der keiner bekannten Bedingung entspricht.

Standardaktion: Die Task wird abnormal beendet.

26 ITEMERR

Tritt in einer der folgenden Situationen auf:

- Die in einem Befehl WRITEQ TS mit der Option REWRITE angegebene Elementnummer ist ungültig (liegt also außerhalb des Bereichs der Eintragsnummern, die für die Warteschlange zugewiesen sind).
- Die maximale Anzahl von Elementen (32767) wurde überschritten.

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

Tritt in einer der folgenden Situationen auf:

- Die Länge der gespeicherten Daten ist Null oder ein negativer Wert.
- Die Länge der gespeicherten Daten ist größer als 32763.

Standardaktion: Die Task wird abnormal beendet.

100 LOCKED

RESP2-Werte:

0

Die Anforderung kann nicht ausgeführt werden, weil die Nutzung der Warteschlange aufgrund des unbestätigten Fehlschlagens einer Arbeitseinheit eingeschränkt wurde.

Standardaktion: Die Task wird abnormal beendet.

18 NOSPACE

Tritt auf, wenn die Option NOSUSPEND angegeben ist und an den folgenden Positionen kein Speicherplatz für die Daten verfügbar ist:

- Hauptspeicher
- Datei im temporären Hilfspuffer
- Listenstruktur des temporären Speicherpools

Diese Bedingung tritt ebenfalls auf, wenn kein Speicherplatz verfügbar ist und es einen aktiven Befehl HANDLE CONDITION für NOSPACE gibt.

Standardaktion: Die Bedingung wird ignoriert.

70 NOTAUTH

RESP2-Werte:

101

Eine Ressourcensicherheitsprüfung ist für QUEUE(name) fehlgeschlagen.

Standardaktion: Die Task wird abnormal beendet.

44 QIDERR

Tritt auf, wenn die durch einen Befehl WRITEQ TS mit Option REWRITE angegebene Warteschlange an den folgenden Positionen nicht zu finden ist:

- Hauptspeicher
- Zusatzspeicher
- Temporärer Speicherpool

Standardaktion: Die Task wird abnormal beendet.

53 SYSIDERR

RESP2-Werte:

4

Tritt in einer der folgenden Situationen auf:

- Die Option SYSID gibt einen Namen an, bei dem es sich nicht um das lokale System oder um ein fernes System handelt (das CICS durch eine CONNECTION- oder IPCONN-Definition bekanntgegeben wird).
- Bei Verwendung der IPIC-Konnektivität handelt es sich beim lokalen und/oder beim fernen System nicht um eine Region von CICS TS 4.2 oder höher.
- Die Verbindung zum fernen System wurde geschlossen.
- Die CICS-Region, in der der Befehl für den temporären Speicher ausgeführt wird, kann keine Verbindung zum TS-Server herstellen, der den TS-Pool verwaltet, von dem die referenzierte Warteschlange für temporären Speicher unterstützt wird. Diese Situation kann beispielsweise vorliegen, wenn die CICS-Region nicht für den Zugriff auf den Server für temporären Speicher berechtigt ist.

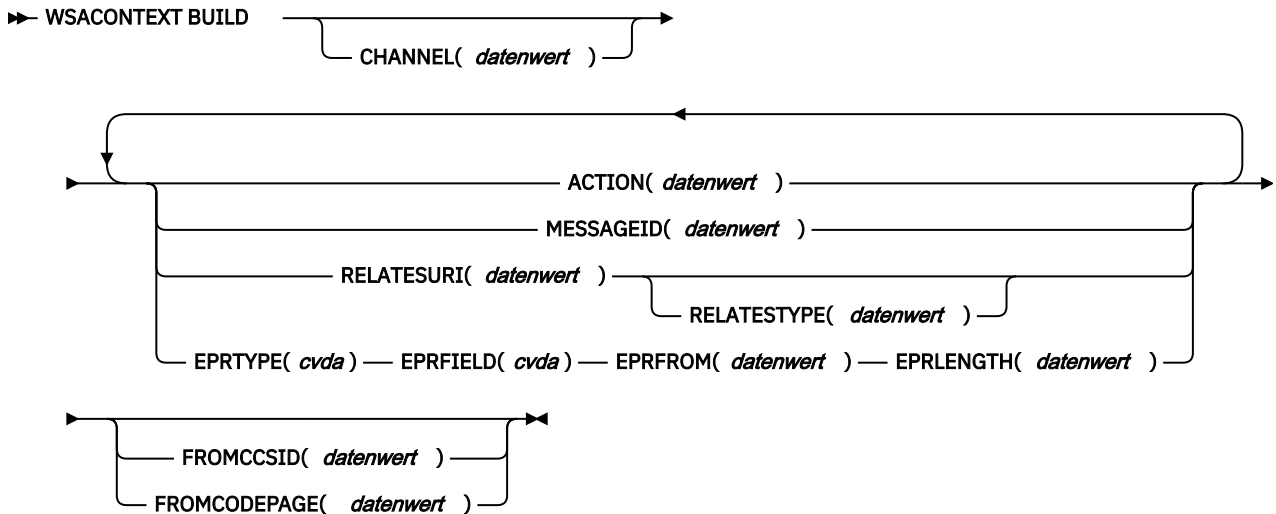
Sie kann ebenfalls vorliegen, falls der Server für temporären Speicher nicht gestartet wurde oder weil der Server fehlgeschlagen ist (bzw. gestoppt wurde), während CICS weiterhin ausgeführt wird.

Standardaktion: Die Task wird abnormal beendet.

WSACONTEXT BUILD

Mit dem Befehl **WSACONTEXT BUILD** können Sie Eigenschaften für die Nachrichtenadressierung (Message Addressing Properties, MAPs) von WS-Addressing im Adressierungskontext einfügen oder ersetzen.

WSACONTEXT BUILD



Bedingungen: CCSIDERR, CHANNELERR, CODEPAGEERR, INVREQ, LENGERR

Dieser Befehl ist threadsicher.

Beschreibung

Verwenden Sie den Befehl **WSACONTEXT BUILD** für eine der folgenden Aktionen:

- MAPs für die Aktions- oder Nachrichten-ID einfügen bzw. ersetzen
- MAPs für Endpunktreferenzen 'To' (= Empfänger), 'From' (= Absender), 'ReplyTo' (= Antwortempfänger) oder 'FaultTo' (Fehlerempfänger) einfügen bzw. ersetzen
- MAPs für 'RelatesTo' einfügen

Sie können den Befehl wiederholt einsetzen, um in jedem Aufruf unterschiedliche Daten bereitzustellen, beispielsweise verschiedene Endpunktreferenzen (EPRs). Die MAPs werden auf alle abgehenden SOAP-Nachrichten angewendet, die mit den Befehlen **INVOKE SERVICE** oder **INVOKE WEBSERVICE** erstellt werden, sowie auf SOAP-Antwortnachrichten von einem Service-Provider.

Optionen

ACTION(datenwert)

Gibt einen Eingabewert an, der eine Aktions-MAP der SOAP-Anforderungsnachricht oder SOAP-Antwortnachricht enthält, z. B. `http://example.ibm.com/namespace/bookingInterface/Make-Booking`. Aktionen werden in den WSDL-Daten bereitgestellt oder durch den Web-Service-Assistenten berechnet, können jedoch durch diese Option überschrieben werden. Der Datenwert muss 255 Zeichen lang sein. Falls die Aktions-MAP kürzer als 255 Zeichen ist, müssen Sie den Datenwert mit nachgestellten Leerzeichen auffüllen.

CHANNEL(datenwert)

Gibt den Namen des Kanals an, der den Adressierungskontext enthält. Der Name des Kanals kann bis zu 16 Zeichen lang sein. Falls der Kanalname kürzer als 16 Zeichen ist, müssen Sie den Datenwert mit nachgestellten Leerzeichen auffüllen. Falls Sie diese Option nicht angeben, wird vom aktuellen Kanal ausgegangen.

Zulässige Zeichen für den Kanalnamen sind A-Z a-z 0-9 \$ @ # / % & ? ! : | " = , ; < > . - und _ . Führende und eingebettete Leerzeichen sind nicht zulässig. Die Gruppe der zulässigen Zeichen für Kanalnamen enthält einige Zeichen, die nicht in allen EBCDIC-Codepages dieselbe Darstellung besitzen. Falls Kanäle zwischen Regionen verlagert werden sollen, empfiehlt es sich daher, bei ihrer Benennung ausschließlich die Zeichen A-Z a-z 0-9 & : = , ; < > . - und _ zu verwenden.

Sie können den Kanalnamen DFHTRANSACTION angeben, um den Transaktionskanal zu verwenden.

Diese Option ist erforderlich, wenn der Befehl **WSACONTEXT BUILD** von einer Anwendung verwendet wird, bei der es sich um den Anforderer eines Web-Service handelt, und gibt den Kanalnamen an, der durch nachfolgende Befehle **INVOKE SERVICE** verwendet wird.

EPRFIELD (cvda)

Gibt das Feld für die Endpunktreferenz an. Sie können diese Option mehrmals verwenden, um eine vollständige Endpunktreferenz zu erstellen.

ADDRESS

In der Option EPRFROM ist das Adressfeld der Endpunktreferenz als URI angegeben.

ALL

In der Option EPRFROM ist eine in XML beschriebene vollständige Endpunktreferenz angegeben.

METADATA

In der Option EPRFROM ist der in XML beschriebene Abschnitt 'Metadata' der Endpunktreferenz angegeben.

REFPARMS

In der Option EPRFROM ist der in XML beschriebene Abschnitt 'ReferenceParameters' der Endpunktreferenz angegeben.

EPRFROM (datenwert)

Ein Eingabedatenwert, der eine vollständige oder partielle Endpunktreferenz enthält, die in den Adressierungskontext eingefügt werden soll. Die Option EPRFIELD beschreibt, welcher Teil der Endpunktreferenz in dieser Option durch die Anwendung angegeben wird. Die Endpunktreferenz kann durch eine Web-Service-Anwendung, die den Befehl **WSAEPR CREATE** verwendet, oder aus einer anderen Quelle bereitgestellt werden, beispielsweise aus einer Service-Registry.

Falls die Option EPRFROM eine Adresse enthält, werden alle Sonderzeichen in der Adresse automatisch mit Escapezeichen versehen oder ohne Escapezeichen verwendet.

Die folgenden Zeichen in einer Adresse werden durch eine Escapezeichenfolge ersetzt:

" , > , < , ' , &

EPRLNGTH (datenwert)

Ein Vollwort-Binärzahlwert für die Eingabedaten, der die Länge der bereitgestellten Endpunktreferenz enthält.

EPRTYPE (cvda)

Gibt den Typ der erstellten Endpunktreferenz an:

TOEPR

Die Zielpunktreferenz, an die eine SOAP-Nachricht gesendet wird.

REPLYTOEPR

Eine Endpunktreferenz, an die eine SOAP-Antwortnachricht zurückgegeben wird.

FAULTTOEPR

Eine Endpunktreferenz, an die eine SOAP-Fehlernachricht zurückgegeben wird.

FROMEPR

Eine Endpunktreferenz, die den Absender der SOAP-Nachricht darstellt.

FROMCCSID (datenwert)

Gibt als Vollwort-Binärzahl die aktuelle ID des codierten Zeichensatzes (Coded Character Set Identifier, CCSID) für die Zeichendaten an, die in den Adressierungskontext eingefügt werden sollen. Falls Sie für die Codepage einen IANA-Namen oder die CCSID mit alphanumerischen Zeichen angeben wollen, verwenden Sie stattdessen die Option FROMCODEPAGE.

Bei Anwendungen von CICS Transaction Server for z/OS ist die CCSID normalerweise eine EBCDIC-CCSID.

Die Standard-CCSID der Region ist im Systeminitialisierungsparameter **LOCALCCSID** angegeben.

Eine Erläuterung der CCSIDs und eine Liste der von CICS unterstützten CCSIDs finden Sie unter [Von CICS unterstützte Konvertierungen](#).

FROMCODEPAGE (datenwert)

Gibt einen von IANA registrierten Namen in alphanumerischen Zeichen oder eine ID des codierten Zeichensatzes (CCSID) für die aktuelle Codepage der Zeichendaten an, die in den Adressierungskontext eingefügt werden sollen; hierzu werden bis zu 40 alphanumerische Zeichen verwendet, die die entsprechende Interpunktion einbeziehen. Verwenden Sie diese Option anstelle der Option CCSID, wenn Sie einen von IANA registrierten Zeichensatznamen verwenden wollen, was im Header 'Content-Type' für eine HTTP-Anforderung angegeben wird. CICS konvertiert den IANA-Namen in eine CCSID; der anschließende Datenkonvertierungsprozess ist identisch. Verwenden Sie diese Option ebenfalls, wenn Sie die CCSID in alphanumerischen Zeichen und nicht als Vollwort-Binärzahl angeben wollen.

Wenn für eine Codepage ein IANA-Name vorhanden ist und CICS seine Verwendung unterstützt, ist der Name mit der CCSID im Abschnitt [Von CICS unterstützte Konvertierungen](#) aufgeführt.

MESSAGEID (datenwert)

Gibt einen Datenwert für eine URI an, die eine SOAP-Nachricht eindeutig kennzeichnet. Der Datenwert muss 255 Zeichen lang sein. Falls die URI kürzer als 255 Zeichen ist, müssen Sie den Datenbereich mit nachgestellten Leerzeichen auffüllen.

RELATESTYPE (datenwert)

Gibt eine URI an, die den Beziehungstyp zwischen der zu sendenden Nachricht und einer anderen Nachricht kennzeichnet. Der Wert muss 255 Zeichen lang sein. Falls die URI kürzer als 255 Zeichen ist, müssen Sie den Datenwert mit nachgestellten Leerzeichen auffüllen. Die Angabe mehrerer MAPs für 'RelatesTo' ist zulässig. Nachfolgende Aufrufe der Optionen RELATESTYPE und RELATESURI erstellen neue MAPs für 'RelatesTo'.

Falls Sie für diese Option keinen Wert angeben, besitzt die MAP für 'RelatesTo' kein Typattribut in der SOAP-Nachricht und nimmt standardmäßig den Wert `http://www.w3.org/2005/08/addressing/reply` an.

RELATESURI (datenwert)

Gibt eine URI mit der Nachrichten-ID einer Nachricht an, mit der die zu sendende Nachricht zusammenhängt. Der Wert muss 255 Zeichen lang sein. Falls die URI kürzer als 255 Zeichen ist, müssen Sie den Datenwert mit nachgestellten Leerzeichen auffüllen.

Bedingungen

123 CCSIDERR

RESP2-Werte:

- 1** Die in der Option FROMCCSID angegebene CCSID liegt außerhalb des Bereichs gültiger CCSID-Werte.
- 2** Die in der Option FROMCCSID angegebene CCSID und die CCSID des Adressierungskontextes stellen eine nicht unterstützte Kombination dar.
- 4** Mindestens ein Zeichen konnte nicht konvertiert werden. Jedes nicht konvertierte Zeichen wurde in den konvertierten Daten durch ein Leerzeichen ersetzt.
- 5** Bei der Codepagekonvertierung der Adressierungskontextdaten trat ein interner Fehler auf.
- 6** Entweder ist die Textcodierung nicht mit der in der Option FROMCCSID angegebenen CCSID kompatibel oder eines bzw. mehrere Zeichen wurden abgeschnitten.

122 CHANNELERR

RESP2-Werte:

- 1**
Der für die Option CHANNEL angegebene Name enthält ein ungültiges Zeichen oder eine ungültige Zeichenkombination.
- 2**
Der angegebene Kanal wurde nicht gefunden.

125 CODEPAGEERR

RESP2-Werte:

- 1**
Die in der Option FROMCODEPAGE angegebene Codepage wird nicht unterstützt.
- 2**
Die in der Option FROMCCSID angegebene Codepage und die CCSID des Adressierungskontextes stellen eine nicht unterstützte Kombination dar.
- 4**
Mindestens ein Zeichen konnte nicht konvertiert werden. Jedes nicht konvertierte Zeichen wurde in den konvertierten Daten durch ein Leerzeichen ersetzt.
- 5**
Bei der Codepagekonvertierung eines Containers trat ein interner Fehler auf.
- 6**
Entweder ist die Textcodierung nicht mit der in der Option FROMCCSID angegebenen CCSID kompatibel oder eines bzw. mehrere Zeichen wurden abgeschnitten.

16 INVREQ

Die RESP2-Werte für die Bedingung INVREQ und die entsprechenden Nachrichten sind nachfolgend aufgeführt. Falls der EIBRESP2-Wert bei diesem Befehl größer als 100 ist, wird das EIBRESP2-Vollwortfeld als Struktur betrachtet, die zwei Halbwörter enthält. Das niedrigstwertige Halbwort enthält immer die Fehlernummer. Das höchstwertige Halbwort enthält die relative Position in den XML-Daten, an der der Parsing-Fehler auftrat.

RESP2-Werte:

- 4**
Die Option CHANNEL war nicht angegeben. Es gibt keinen aktuellen Kanal, weil das Programm, das den Befehl ausgegeben hat, nicht den Namen eines Kanals übergeben hat.
- 6**
Das Feld ACTION enthält keine gültigen URI-Zeichen.
- 7**
Das Feld MESSAGEID enthält keine gültigen URI-Zeichen.
- 8**
Das Feld RELATESURI enthält keine gültigen URI-Zeichen.
- 9**
Das Feld RELATESTYPE enthält keine gültigen URI-Zeichen.
- 10**
Die Option EPRFROM enthält keine gültigen XML-Daten.
- 13**
Die Option EPRFROM enthält keine gültigen XML-Daten. Der Abschnitt <Metadata> der Endpunktreferenz enthält möglicherweise keine gültigen XML-Daten.
- 14**
Die Option EPRFROM enthält keine gültigen XML-Daten. Der Abschnitt <ReferenceParameters> der Endpunktreferenz enthält möglicherweise keine gültigen XML-Daten.
- 15**
Die Option EPRFROM enthält möglicherweise keine gültige URI.

22 LENGERR

RESP2-Werte:

Diese Bedingung tritt auf, wenn die Länge der gespeicherten Daten den durch die Option EPR-LENGTH angegebenen Wert überschreitet. Sie gilt nur für die Option EPRINTO und kann bei der Option EPRSET nicht auftreten.

Standardaktion: Die Task wird abnormal beendet.

WSACONTEXT DELETE

Mit dem Befehl **WSACONTEXT DELETE** können Sie den Adressierungskontext löschen.

WSACONTEXT DELETE

➡ WSADDCONTEXT DELETE — CHANNEL(*datenwert*) →

Bedingungen: CHANNELERR, NOTFND, INVREQ

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl **WSACONTEXT DELETE** löscht die Adressierungskontexte für die SOAP-Anforderungsnachrichten und SOAP-Antwortnachrichten aus dem angegebenen Kanal. Sie können diesen Befehl nur in einer Anwendung zur Anforderung von Web-Services verwenden.

Optionen

CHANNEL (*datenwert*)

Gibt den Namen des Kanals an, der den Adressierungskontext enthält. Der Name des Kanals kann bis zu 16 Zeichen lang sein. Falls der Kanalname kürzer als 16 Zeichen ist, müssen Sie den Datenwert mit nachgestellten Leerzeichen auffüllen. Sie können den Kanalnamen DFHTRANSACTION angeben, um den Transaktionskanal zu verwenden.

Bedingungen

122 CHANNELERR

RESP2-Werte:

1

Der für die Option CHANNEL angegebene Name enthält ein falsches Zeichen oder eine falsche Zeichenkombination.

2

Der angegebene Kanal wurde nicht gefunden.

16 INVREQ

RESP2-Werte:

5

Dieser Befehl ist bei einem Web-Service-Provider nicht zulässig.

13 NOTFND

RESP2-Werte:

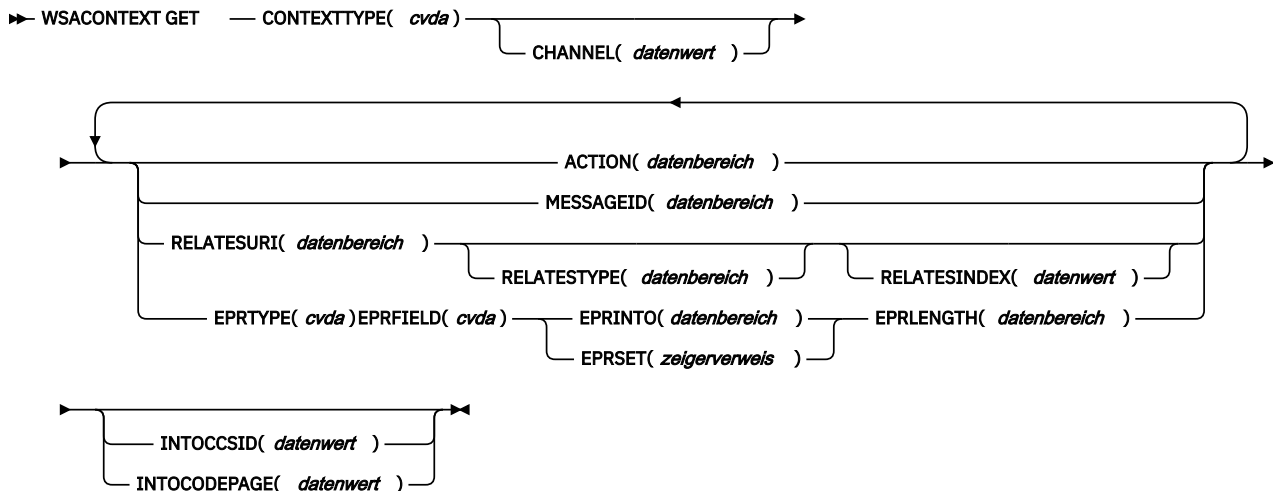
3

Der Adressierungskontext wurde im angegebenen Kanal nicht gefunden.

WSACONTEXT GET

Mit dem Befehl **WSACONTEXT GET** können Sie in einem Service-Provider die Nachrichtenadressierungseigenschaften (Message Addressing Properties, MAPs) abrufen, die vom Serviceanforderer gesendet wurden. In einem Serviceanforderer können Sie mit dem Befehl **WSACONTEXT GET** die MAPs der Antwortnachricht abrufen. Die Verwendung dieses Befehls durch Axis2-Anwendungen, die in einer Axis2-Pipeline in CICS gehostet werden, ist nicht möglich.

WSACONTEXT GET



Bedingungen: CCSIDERR, CHANNELERR, CODEPAGEERR, INVREQ, LENGERR, NOTFND

Dieser Befehl ist threadsicher.

Beschreibung

Mit dem Befehl **WSACONTEXT GET** werden in einem Service-Provider die MAPs des Anforderers aus dem Adressierungskontext abgerufen. In einem Serviceanforderer werden hiermit die MAPs des Providers aus der Antwortnachricht abgerufen. Der Befehl **WSACONTEXT GET** kann wiederholt aufgerufen werden, um verschiedene MAPs zurückzugeben.

Optionen

ACTION(datenbereich)

Gibt einen Ausgabebereich an, der die Aktions-MAP der SOAP-Anforderungsnachricht oder SOAP-Antwortnachricht enthalten soll. Der Datenbereich muss 255 Zeichen lang sein. Falls die Aktions-MAP kürzer als 255 Zeichen ist, füllt CICS den Datenbereich mit nachgestellten Leerzeichen auf.

CHANNEL(datenwert)

Gibt den Namen des Kanals an, der den Adressierungskontext enthält. Der Name des Kanals kann bis zu 16 Zeichen lang sein. Falls der Kanalname kürzer als 16 Zeichen ist, müssen Sie den Datenwert mit nachgestellten Leerzeichen auffüllen. Falls Sie diese Option nicht angeben, wird vom aktuellen Kanal ausgegangen.

Zulässige Zeichen für den Kanalnamen sind A-Z a-z 0-9 \$ @ # / % & ? ! : | " = , ; < > . - und _ . Führende und eingebettete Leerzeichen sind nicht zulässig. Die Gruppe der zulässigen Zeichen für Kanalnamen enthält einige Zeichen, die nicht in allen EBCDIC-Codepages dieselbe Darstellung besitzen. Falls Kanäle zwischen Regionen verlagert werden sollen, empfiehlt es sich daher, bei ihrer Benennung ausschließlich die Zeichen A-Z a-z 0-9 & : = , ; < > . - und _ zu verwenden.

Sie können den Kanalnamen DFHTRANSACTION angeben, um den Transaktionskanal zu verwenden.

Diese Option ist erforderlich, wenn der Befehl **WSACONTEXT GET** von einer Anwendung verwendet wird, bei der es sich um den Anforderer eines Web-Service handelt, und gibt den Kanalnamen an, der durch nachfolgende Befehle **INVOKE SERVICE** verwendet wird.

CONTEXTTYPE(cvda)

Gibt an, aus welchem Adressierungskontexttyp die MAPs abgerufen werden sollen. Die CVDA-Werte lauten wie folgt:

REQCONTEXT

Der Adressierungskontext, der die Anforderung enthält. Auf die MAPs in diesem Adressierungskontext kann entweder ein Web-Service-Anforderer oder eine Anwendung zugreifen, bei der es sich um einen Web-Service-Provider handelt.

RESPCONTEXT

Der Adressierungskontext, der die Antwort enthält. Auf die MAPs in diesem Adressierungskontext kann ausschließlich eine Anwendung für einen Web-Service-Anforderer zugreifen.

EPRFIELD(*cvda*)

Gibt den Teil der Endpunktreferenz an, der in dem durch EPRINTO angegebenen Datenbereich zurückgegeben werden soll. Die CVDA-Werte lauten wie folgt:

ADDRESS

Gibt das Adressfeld der Endpunktreferenz zurück.

ALL

Gibt die vollständige Endpunktreferenz in XML zurück.

METADATA

Gibt den Abschnitt 'Metadata' der Endpunktreferenz in XML zurück.

REFPARMS

Gibt den Abschnitt 'ReferenceParameters' der Endpunktreferenz in XML zurück.

EPRINTO(*datenbereich*)

Ein Ausgabedatenbereich, der für die vollständige oder partielle Endpunktreferenz verwendet wird. Die Optionen EPRINTO und EPRSET schließen sich gegenseitig aus.

Falls die Option EPRINTO eine Adresse enthält, werden alle Sonderzeichen in der Adresse automatisch mit Escapezeichen versehen oder ohne Escapezeichen verwendet.

Die folgenden Zeichen in einer Adresse werden durch eine Escapezeichenfolge ersetzt:

" , > , < , ' , &

ERPLENGTH(*datenbereich*)

Gibt als Halbwort-Binärwert die Länge der Endpunktreferenz an.

Falls Sie die Option EPRINTO angeben, müssen Sie einen Wert für ERPLENGTH angeben, sofern die Länge nicht vom Compiler aus der Option EPRINTO generiert werden kann.

Die Option ERPLENGTH definiert die maximale Länge der Daten, die vom Programm akzeptiert wird. Falls der angegebene Wert kleiner als Null ist, wird Null angenommen. Falls die Länge der Daten den angegebenen Wert überschreitet, werden die Daten auf diesen Wert abgeschnitten und die Bedingung LENGERR tritt auf.

EPRSET(*zeigerverweis*)

Gibt den Zeigerverweis an, der auf die Adresse des Ausgabedatenbereichs gesetzt wird, der die vollständige oder partielle Endpunktreferenz enthält. Sofern der Zeigerverweis nicht durch andere Befehle oder Anweisungen geändert wird, bleibt er bis zum nächsten Befehl **WSACONTEXT GET** oder bis zum Ende der Task gültig. Die Optionen EPRINTO und EPRSET schließen sich gegenseitig aus.

EPRTYPE(*cvda*)

Gibt den Typ der angeforderten Endpunktreferenz an. Die CVDA-Werte lauten wie folgt:

TOEPR

Die Zielpunktreferenz, an die eine SOAP-Nachricht gesendet wird.

REPLYTOEPR

Eine Endpunktreferenz, an die eine SOAP-Antwortnachricht zurückgegeben wird.

FAULTTOEPR

Eine Endpunktreferenz, an die eine SOAP-Fehlernachricht zurückgegeben wird.

FROMEPR

Eine Endpunktreferenz, die den Absender der SOAP-Nachricht darstellt.

INTOCCSID(*datenwert*)

Gibt als Vollwort-Binärzahl die ID des codierten Zeichensatzes (Coded Character Set Identifier, CCSID) an, in die die Zeichendaten im Adressierungskontext konvertiert werden sollen. Falls Sie für die Codepage einen IANA-Namen oder die CCSID mit alphanumerischen Zeichen angeben wollen, verwenden Sie stattdessen die Option INTOCODEPAGE.

Bei Anwendungen von CICS Transaction Server for z/OS ist die CCSID normalerweise eine EBCDIC-CCSID.

Eine Erläuterung der CCSIDs und eine Liste der von CICS unterstützten CCSIDs finden Sie unter [Von CICS unterstützte Konvertierungen](#).

INTOCODEPAGE (datenwert)

Gibt einen von IANA registrierten Namen in alphanumerischen Zeichen oder eine ID des codierten Zeichensatzes (CCSID) für die Codepage an, in die die Zeichendaten im Adressierungskontext konvertiert werden sollen; hierzu werden bis zu 40 alphanumerische Zeichen verwendet, die die entsprechende Interpunktion einbeziehen. Verwenden Sie diese Option anstelle der Option CCSID, wenn Sie einen von IANA registrierten Zeichensatznamen verwenden wollen, was im Header 'Content-Type' für eine HTTP-Anforderung angegeben wird. CICS konvertiert den IANA-Namen in eine CCSID. Der nachfolgende Datenkonvertierungsprozess ist identisch. Verwenden Sie diese Option außerdem dann, wenn Sie es bevorzugen, die CCSID in alphanumerischen Zeichen und nicht als Vollwort-Binärwert anzugeben.

Wenn für eine Codepage ein IANA-Name vorhanden ist und CICS seine Verwendung unterstützt, ist der Name mit der CCSID im Abschnitt [Von CICS unterstützte Konvertierungen](#) aufgeführt.

MESSAGEID (datenbereich)

Gibt einen Ausgabebereich an, der die MAP 'MessageID' der SOAP-Anforderungsnachricht oder SOAP-Antwortnachricht enthalten soll. Der Datenbereich muss 255 Zeichen lang sein. Falls die MAP 'MessageID' kürzer als 255 Zeichen ist, füllt CICS den Datenbereich mit nachgestellten Leerzeichen auf.

RELATESINDEX (datenwert)

Gibt einen numerischen Wert an, der festlegt, welche MAP für 'RelatesTo' zurückgegeben werden soll. Falls dieser Wert nicht angegeben wird, wird die erste MAP für 'RelatesTo' zurückgegeben. Der Index beginnt bei 1. Falls der Wert des Index größer als die Anzahl der MAPs für 'RelatesTo' ist, werden in den Datenbereichen von RELATESTYPE und RELATESURI Leerzeichen zurückgegeben.

RELATESTYPE (datenbereich)

Gibt einen Ausgabebereich mit einer URI an, die den Beziehungstyp zwischen dieser Nachricht und einer anderen Nachricht kennzeichnet. Der Datenbereich muss 255 Zeichen lang sein. Falls die URI kürzer als 255 Zeichen ist, füllt CICS den Datenbereich mit nachgestellten Leerzeichen auf.

RELATESURI (datenbereich)

Gibt einen Ausgabebereich mit der URI der MAP für 'RelatesTo' mit dem Beziehungstyp zwischen dieser Nachricht und einer anderen Nachricht an. Der Datenbereich muss 255 Zeichen lang sein. Falls die MAP 'MessageID' kürzer als 255 Zeichen ist, füllt CICS den Datenbereich mit nachgestellten Leerzeichen auf.

Bedingungen

123 CCSIDERR

RESP2-Werte:

1

Die in der Option INTOCCSID angegebene CCSID liegt außerhalb des Bereichs gültiger CCSID-Werte.

2

Die in der Option FROMCCSID angegebene CCSID und die CCSID des Adressierungskontextes stellen eine nicht unterstützte Kombination dar.

4

Mindestens ein Zeichen konnte nicht konvertiert werden. Jedes nicht konvertierte Zeichen wurde in den konvertierten Daten durch ein Leerzeichen ersetzt.

5

Bei der Codepagekonvertierung der Adressierungskontextdaten trat ein interner Fehler auf.

122 CHANNELERR

RESP2-Werte:

1
Der für die Option CHANNEL angegebene Name enthält ein falsches Zeichen oder eine falsche Zeichenkombination.

2
Der angegebene Kanal wurde nicht gefunden.

125 CODEPAGEERR

RESP2-Werte:

1
Die in der Option INTOCODEPAGE angegebene Codepage wird nicht unterstützt.

2
Die in der Option INTOCODEPAGE angegebene Codepage und die CCSID des Adressierungskontextes stellen eine nicht unterstützte Kombination dar.

4
Mindestens ein Zeichen konnte nicht konvertiert werden. Jedes nicht konvertierte Zeichen wurde in den konvertierten Daten durch ein Leerzeichen ersetzt.

5
Bei der Codepagekonvertierung eines Containers trat ein interner Fehler auf.

16 INVREQ

RESP2-Werte:

4
Die Option CHANNEL war nicht angegeben. Es gibt keinen aktuellen Kanal, weil das Programm, das den Befehl ausgegeben hat, nicht den Namen eines Kanals übergeben hat.

11
Die Option RELATESINDEX ist ungültig.

12
Der Wert für die Option RELATESINDEX ist größer als die Anzahl der MAPs für 'RelatesTo'.

22 LENGERR

RESP2-Werte:

20
Diese Bedingung tritt auf, wenn die Länge der gespeicherten Daten den durch die Option EPR-LENGTH angegebenen Wert überschreitet. Sie gilt nur für die Option EPRINTO und kann bei der Option EPRSET nicht auftreten.

Standardaktion: Die Task wird abnormal beendet.

13 NOTFND

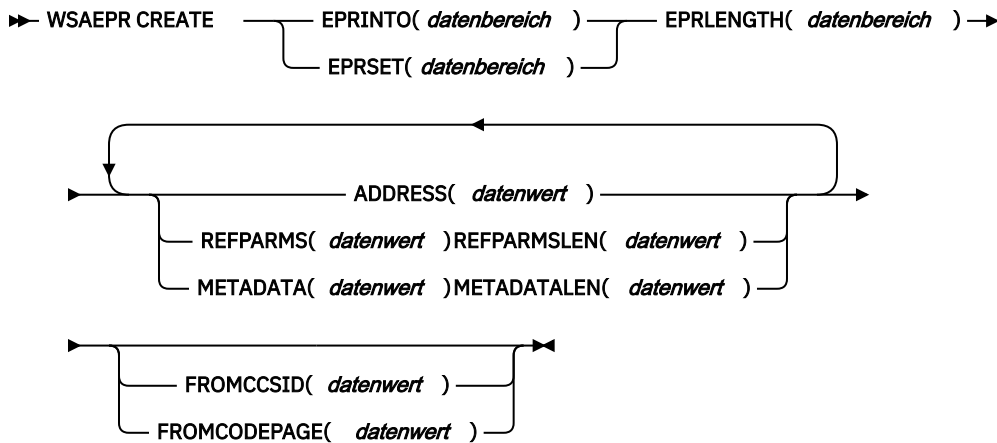
RESP2-Werte:

3
Der Adressierungskontext wurde nicht gefunden.

WSAEPR CREATE

Mit dem Befehl **WSAEPR CREATE** können Sie eine Endpunktreferenz (EPR) erstellen, um einen Web-Service oder eine Web-Service-Ressource darzustellen.

WSAEPR CREATE



Bedingungen: CCSIDERR, CODEPAGEERR, INVREQ, LENGERR

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl **WSAEPR CREATE** erstellt eine Endpunktreferenz, die einen Web-Service oder eine Web-Service-Ressource darstellen kann. Diese Endpunktreferenz können Sie an einen Client senden, damit der Adressierungskontext für Anforderungen des Service verwendet wird.

Optionen

ADDRESS(*datenwert*)

Gibt die URI an, die die Adresse der Endpunktreferenz bildet. Der Wert dieser Option muss 255 Zeichen lang sein. Falls die URI kürzer als 255 Zeichen ist, müssen Sie den Datenwert mit nachgestellten Leerzeichen auffüllen.

Die Standardadresse `http://www.w3.org/2005/08/addressing/anonymous` gibt Informationen an das aufrufende Programm des Befehls zurück. Verwenden Sie die Adresse `http://www.w3.org/2005/08/addressing/none`, falls keine Anforderung oder Antwort erforderlich ist. Wenn die Endpunktreferenz 'To' eine URI enthält, wird die SOAP-Nachricht an diese URI gesendet. Wenn die Endpunktreferenzen 'ReplyTo' oder 'FaultTo' eine URI enthalten, werden Antwortnachrichten unter Verwendung dieser URI an den Web-Service und nicht zurück an den Absender der Anforderungsnachricht gesendet.

EPRINTO(*datenbereich*)

Gibt den Datenbereich an, der die generierte Endpunktreferenz enthalten soll. Die Optionen EPRINTO und EPRSET schließen sich gegenseitig aus.

Falls die Option EPRINTO eine Adresse enthält, werden alle Sonderzeichen in der Adresse automatisch mit Escapezeichen versehen oder ohne Escapezeichen verwendet.

Die folgenden Zeichen in einer Adresse werden durch eine Escapezeichenfolge ersetzt:

" , > , < , ' , &

EPRLENGTH(*datenbereich*)

Gibt als Halbwort-Binärwert die Länge der Endpunktreferenz an.

Falls Sie die Option EPRINTO angeben, müssen Sie einen Wert für EPRLENGTH angeben, sofern die Länge nicht vom Compiler aus der Option EPRINTO generiert werden kann.

Die Option EPRLENGTH definiert die maximale Länge der Daten, die vom Programm akzeptiert wird. Falls der angegebene Wert kleiner als Null ist, wird Null angenommen. Falls die Länge der Daten den angegebenen Wert überschreitet, werden die Daten auf diesen Wert abgeschnitten und die Bedingung LENGERR tritt auf.

EPRSET (zeigerverweis)

Gibt den Zeigerverweis an, der auf die Adresse des Ausgabedatenbereichs gesetzt wird, der die vollständige oder partielle Endpunktreferenz enthält. Sofern der Zeigerverweis nicht durch andere Befehle oder Anweisungen geändert wird, bleibt er bis zum nächsten Befehl **WSACONTEXT GET** oder bis zum Ende der Task gültig. Die Optionen EPRINTO und EPRSET schließen sich gegenseitig aus.

FROMCCSID (datenwert)

Gibt als Vollwort-Binärzahl die aktuelle ID des codierten Zeichensatzes (Coded Character Set Identifier, CCSID) für die zu lesenden oder zu schreibenden Zeichendaten an. Falls Sie für die Codepage einen IANA-Namen oder die CCSID mit alphanumerischen Zeichen angeben wollen, verwenden Sie stattdessen die Option FROMCODEPAGE.

Bei Anwendungen von CICS Transaction Server for z/OS ist die CCSID normalerweise eine EBCDIC-CCSID.

Die Standard-CCSID der Region ist in der Systeminitialisierungsoption LOCALCCSID angegeben.

Eine Erläuterung der CCSIDs und eine Liste der von CICS unterstützten CCSIDs finden Sie unter [Von CICS unterstützte Konvertierungen](#).

FROMCODEPAGE (datenwert)

Gibt einen von IANA registrierten Namen in alphanumerischen Zeichen oder eine ID des codierten Zeichensatzes (CCSID) für die aktuelle Codepage der zu lesenden oder zu schreibenden Zeichendaten an; hierzu werden bis zu 40 alphanumerische Zeichen verwendet, die die entsprechende Interpunktion einbeziehen. Verwenden Sie diese Option anstelle der Option CCSID, wenn Sie einen von IANA registrierten Zeichensatznamen verwenden wollen, was im Header 'Content-Type' für eine HTTP-Anforderung angegeben wird. CICS konvertiert den IANA-Namen in eine CCSID; der anschließende Datenkonvertierungsprozess ist identisch. Verwenden Sie diese Option ebenfalls, wenn Sie die CCSID in alphanumerischen Zeichen und nicht als Vollwort-Binärzahl angeben wollen.

Wenn für eine Codepage ein IANA-Name vorhanden ist und CICS seine Verwendung unterstützt, ist der Name mit der CCSID im Abschnitt [Von CICS unterstützte Konvertierungen](#) aufgeführt.

METADATA (datenwert)

Gibt Metadaten an, die das Verhalten, die Richtlinien und die Leistungsmerkmale des Endpunktes beschreiben, auf den sich die Endpunktreferenz bezieht. Die Metadaten müssen in XML beschrieben sein.

METADATALEN (datenwert)

Gibt als Vollwort-Binärwert die Länge der Option METADATA an.

REFPARMS (datenwert)

Gibt Optionen für die Anwendungsreferenz an, die einen Teil der Endpunktreferenz bilden. Diese Optionen sind in XML beschrieben.

REFPARMSLEN (datenwert)

Gibt als Vollwort-Binärwert die Länge der Referenzoptionen an.

Bedingungen**123 CCSIDERR**

RESP2-Werte:

1

Die in der Option FROMCCSID angegebene CCSID liegt außerhalb des Bereichs gültiger CCSID-Werte.

2

Die in der Option FROMCCSID angegebene CCSID und die CCSID des Containers stellen eine nicht unterstützte Kombination dar.

5

Bei der Codepagekonvertierung eines Containers trat ein interner Fehler auf.

- 6** Entweder ist die Textcodierung nicht mit der in der Option FROMCCSID angegebenen CCSID kompatibel oder eines bzw. mehrere Zeichen wurden abgeschnitten.

125 CODEPAGEERR

RESP2-Werte:

- 1** Die in der Option FROMCODEPAGE angegebene Codepage wird nicht unterstützt.
- 2** Die in der Option FROMCCSID angegebene Codepage und die CCSID des Containers stellen eine nicht unterstützte Kombination dar.
- 5** Bei der Codepagekonvertierung eines Containers trat ein interner Fehler auf.
- 6** Entweder ist die Textcodierung nicht mit der in der Option FROMCODEPAGE angegebenen Codepage kompatibel oder eines bzw. mehrere Zeichen wurden abgeschnitten.

16 INVREQ

Die RESP2-Werte für die Bedingung INVREQ und die entsprechenden Nachrichten sind nachfolgend aufgeführt. Falls der EIBRESP2-Wert bei diesem Befehl größer als 100 ist, wird das EIBRESP2-Vollwortfeld als Struktur betrachtet, die zwei Halbwörter enthält. Das niedrigstwertige Halbwort enthält immer die Fehlernummer. Das höchstwertige Halbwort enthält die relative Position in den XML-Daten, an der der Parsing-Fehler auftrat.

RESP2-Werte:

- 8** Einer der Eingabeparameter enthält einen falschen Wert. Die Option ADDRESS enthält möglicherweise keine gültige URI.
- 13** Einer der Eingabeparameter enthält einen falschen Wert. Die Option METADATA enthält möglicherweise keine gültigen XML-Daten.
- 14** Einer der Eingabeparameter enthält einen falschen Wert. Die Option REFPARMS enthält möglicherweise keine gültigen XML-Daten.

22 LENGERR

RESP2-Werte:

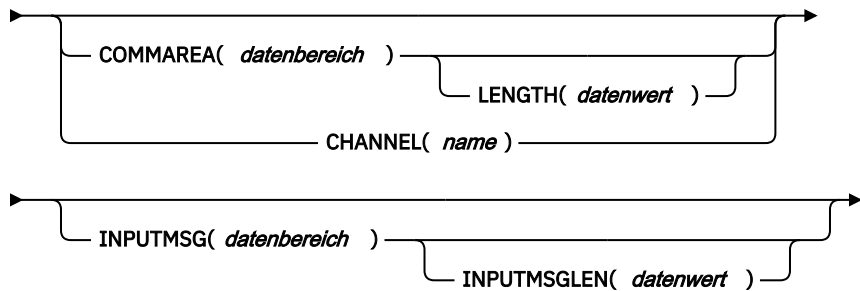
- 20** Diese Bedingung tritt auf, wenn die Länge der gespeicherten Daten den durch die Option EPRLENGTH angegebenen Wert überschreitet. Sie gilt nur für die Option EPRINTO und kann bei der Option EPRSET nicht auftreten.
- Standardaktion: Die Task wird abnormal beendet.

XCTL

Überträgt die Programmsteuerung.

XCTL

➔ XCTL — PROGRAM(*name*) ➔



Bedingungen: CHANNELERR, INVREQ, LENGERR, NOTAUTH, PGMIDERR

Dieser Befehl ist threadsicher.

Beschreibung

Der Befehl XCTL überträgt die Steuerung von einem Anwendungsprogramm an ein anderes Anwendungsprogramm auf derselben logischen Ebene. Das Programm, von dem die Steuerung übertragen wird, wird freigegeben. Falls sich das Programm, an das die Steuerung übertragen wird, noch nicht im Hauptspeicher befindet, wird es geladen.

Dieser Befehl wird im aktuellen Anwendungskontext ausgeführt. Falls er durch ein Programm ausgegeben wird, das unter einer Task für eine auf einer Plattform implementierte Anwendung ausgeführt wird, sucht CICS im privaten Programmverzeichnis für die Anwendung nach dem benannten Programm. Dieser Befehl kann nicht zur Änderung des Anwendungskontextes verwendet werden.

Anmerkung: Die Verwendung des Befehls XCTL zum Aufrufen einer Java EE-Anwendung, die in einem Liberty-JVM-Server ausgeführt wird, wird nicht unterstützt. Sie können eine Verbindung zu einer in einem Liberty-JVM-Server ausgeführten Java EE-Anwendung entweder als Startprogramm einer CICS-Transaktion oder mithilfe der Befehle LINK, START oder START CHANNEL aus einem beliebigen CICS-Programm herstellen. Weitere Informationen finden Sie unter [Verknüpfung zu einer Java EE- oder einer Spring Boot-Anwendung aus einem CICS-Programm herstellen](#).

Optionen

CHANNEL(name)

Gibt den 1 bis 16 Zeichen langen Namen eines Kanals an, der für das aufgerufene Programm verfügbar gemacht werden soll. Zulässige Zeichen: A-Z a-z 0-9 \$ @ # / % & ? ! : | " = ~ , ; < > . - und _ . Führende und eingebettete Leerzeichen sind nicht zulässig. Falls der angegebene Name weniger als 16 Zeichen umfasst, wird er mit nachgestellten Leerzeichen auf 16 Zeichen aufgefüllt. Wenn der Kanal nicht vorhanden ist, wird er erstellt. Dieser neue Kanal bleibt im Bereich, bis die Verbindungsebene geändert wird. Informationen zum Geltungsbereich von Kanälen finden Sie unter [Geltungsbereich eines Kanals](#).

Kanalnamen haben immer EBCDIC-Format. Die oben angegebene Gruppe der zulässigen Zeichen für Kanalnamen enthält einige Zeichen, die nicht in allen EBCDIC-Codepages dieselbe Darstellung besitzen. Falls Kanäle zwischen Regionen verlagert werden sollen, empfiehlt es sich daher, bei ihrer Benennung ausschließlich die Zeichen A-Z a-z 0-9 & : = , ; < > . - und _ zu verwenden.

Sie können den Kanalnamen DFHTRANSACTION angeben, um einen Transaktionskanal zu verwenden. Ein Transaktionskanal verliert nicht seine Gültigkeit, wenn sich die Verbindungsebene ändert; der Zugriff auf diesen Kanal ist in der Task immer möglich. Weitere Informationen finden Sie unter [Kanäle und Container](#).

Das Programm, das den Befehl XCTL ausgibt, kann Folgendes vornehmen:

- Es kann mit einem oder mehreren Befehlen **PUT CONTAINER CHANNEL** bzw. **PUT64 CONTAINER** den Kanal erstellen, bevor es den Befehl **XCTL** ausgibt.

- Es kann seinen aktuellen Kanal durch den Namen angeben.
- Es kann einen Kanal benennen, der zurzeit nicht vorhanden ist. In diesem Fall wird ein neuer leerer Kanal erstellt.

COMMAREA(datenbereich)

Gibt einen Kommunikationsbereich an, der für das aufgerufene Programm verfügbar gemacht werden soll. In dieser Option wird der Inhalt des Datenbereichs übergeben. Bei COBOL müssen Sie diesem Datenbereich im empfangenden Programm den Namen DFHCOMMAREA zuweisen. Weitere Informationen finden Sie unter [Daten an andere Programme übergeben](#).

INPUTMSG(datenbereich)

Gibt die Daten an, die an das aufgerufene Programm übergeben werden sollen, wenn es erstmalig einen Befehl RECEIVE ausgibt. Falls das aufgerufene Programm mit einem Befehl **LINK** die Steuerung an ein anderes Programm übergibt, wird eine verknüpfte Kette erstellt (eine Beschreibung finden Sie im Abschnitt über die Option INPUTMSG des Befehls **LINK**). Die in INPUTMSG angegebenen Daten bleiben verfügbar, bis ein Befehl **RECEIVE** ausgegeben oder die Steuerung an CICS zurückgegeben wird.

INPUTMSGLEN(datenwert)

Gibt als Halbwort-Binärwert die Länge der Daten an, die durch die Option INPUTMSG übergeben werden.

LENGTH(datenwert)

Gibt als Halbwort-Binärdatenwert die Länge des Kommunikationsbereichs in Byte an. Eine Beschreibung für eine angemessene Obergrenze finden Sie unter [„LENGTH-Optionen in CICS-Befehlen“](#) auf Seite 10.

PROGRAM(name)

Gibt die 1 bis 8 alphanumerische Zeichen lange Kennung des Programms an, an das die Steuerung unbedingt übergeben werden soll. Der angegebene Name muss für CICS als Programm definiert sein; falls AUTOINSTALL aktiv ist, wird jedoch eine Definition automatisch installiert.

Die Verwendung von Anführungszeichen ist zu beachten:

```
EXEC CICS XCTL PROGRAM('PROGX')
```

PROGX steht in Anführungszeichen, weil es sich um den Programmnamen handelt.

```
EXEC CICS XCTL PROGRAM(DAREA)
```

DAREA steht nicht in Anführungszeichen, weil dies der Name eines Datenbereichs ist, der den tatsächlichen Programmnamen enthält. Wenn für den Programmnamen ein Datenbereich verwendet wird, muss dieser Datenbereich als acht Byte langes Feld im Arbeitsspeicher definiert werden.

Bedingungen

122 CHANNELERR

RESP2-Werte:

1

Der in der Option CHANNEL angegebene Name enthält ein ungültiges Zeichen oder eine ungültige Zeichenkombination.

16 INVREQ

RESP2-Werte:

8

Ein Befehl XCTL mit der Option INPUTMSG wurde für ein Programm ausgegeben, das keinem Terminal zugeordnet ist oder das einer logischen APPC-Einheit bzw. einer IRC-Sitzung zugeordnet ist.

29

Der EXEC-Befehl XCTL ist bei GLUE oder TRUE nicht zulässig.

30

Die Programmmanagerdomäne wurde noch nicht initialisiert. Dies liegt wahrscheinlich daran, dass eine Anforderung XCTL in einem PLT-Programm der ersten Phase ausgegeben wurde.

31

Ein Befehl XCTL wurde von einem Programm, das mit einem Anwendungskontext ausgeführt wird, an ein anderes Programm ausgegeben, bei dem es sich um einen Anwendungseinstiegspunkt handelt.

32

Ein Befehl XCTL wurde von einem Programm ausgegeben, das eine Java EE-Anwendung in einem Liberty-JVM-Server aufrufen soll. Diese Operation wird nicht unterstützt.

33

Ein Befehl XCTL wurde aus einem öffentlichen Programm an ein anderes Programm ausgegeben, bei dem es sich um einen Anwendungseinstiegspunkt handelt.

34

Ein Befehl XCTL wurde von einem Programm, das mit einem Anwendungskontext ausgeführt wird, an ein öffentliches Programm ausgegeben.

200

Ein Befehl XCTL wurde mit der Option INPUTMSG in einem durch DPL aufgerufenen Programm ausgegeben.

Standardaktion: Die Task wird abnormal beendet.

22 LENGERR

RESP2-Werte:

11

Der Wert für LENGTH ist kleiner als 0 oder größer als 32763.

26

Als Adresse für COMMAREA wurde Null übergeben, aber der Wert für LENGTH war ungleich null.

27

Der Wert für INPUTMSGLEN war kleiner als 0 oder größer als 32767.

28

Der Wert für LENGTH oder INPUTMSGLEN ist größer als die Länge des in den Optionen COMMAREA oder INPUTMSG angegebenen Datenbereichs und während diese Daten kopiert wurden, trat aufgrund der falschen Länge eine zerstörerische Überschneidung auf.

Standardaktion: Die Task wird abnormal beendet.

70 NOTAUTH

RESP2-Werte:

101

Eine Ressourcensicherheitsprüfung ist für PROGRAM(name) fehlgeschlagen.

Standardaktion: Die Task wird abnormal beendet.

27 PGMIDERR

RESP2-Werte:

1

Für ein Programm ist keine Ressourcendefinition installiert und entweder war die automatische Programminstallation inaktiviert oder das Steuerprogramm zur automatischen Programminstallation gab an, dass das Programm nicht automatisch installiert werden soll.

2

Das Programm ist inaktiviert.

3

Ein Programm konnte aus einem der folgenden Gründe nicht geladen werden:

- Das Programm wurde erstmalig geladen und das Laden des Programms ist fehlgeschlagen; dies liegt in der Regel daran, dass das Lademodul nicht gefunden wurde.
- Es handelte sich um ein nachfolgendes Laden des Programms, aber der erste Ladevorgang war fehlgeschlagen.

Zum Zurücksetzen des Ladestatus muss das Lademodul in der CICS-Programmbibliothek (DFHRPL) oder der dynamischen Bibliotheksverkettung angegeben sein und ein Befehl SET PROGRAM NEWCOPY ausgeführt werden.

9

Die installierte Programmdefinition bezieht sich auf ein fernes Programm.

21

Das Steuerprogramm zur automatischen Programminstallation ist fehlgeschlagen, weil das Steuerprogramm zur automatischen Programminstallation entweder falsch bzw. falsch definiert ist oder abgebrochen wurde. Die automatische Programminstallation ist inaktiviert und die Nachricht DFHPG0202 oder DFHPG0203 wird an CSPL geschrieben.

22

Das vom Steuerprogramm zur automatischen Programminstallation zurückgegebene Modell war für CICS nicht definiert oder war nicht aktiviert.

23

Das Steuerprogramm zur automatischen Programminstallation hat ungültige Daten zurückgegeben.

24

Die Definition für das Programm ist fehlgeschlagen, weil die automatische Installation einen ungültigen Programmnamen oder eine ungültige Definition zurückgegeben hat.

Standardaktion: Die Task wird abnormal beendet.

Beispiele

Das folgende Beispiel zeigt, wie eine Übertragung der Steuerung an ein Anwendungsprogramm namens PROG2 angefordert wird:

```
EXEC CICS XCTL PROGRAM('PROG2')
```

Kapitel 3. Threadsichere Befehle

Hier sind die Befehle aufgelistet, die threadsicher oder unter bestimmten Bedingungen threadsicher sind.

Nicht alle **EXEC CICS**-Befehle sind threadsicher; die Ausgabe von nicht threadsicheren Befehlen bewirkt, dass CICS den QR-Tasksteuerblock verwendet, um die Serialisierung sicherzustellen. Informationen zum Schreiben von threadsicheren Anwendungsprogrammen finden Sie unter [Threadsichere Programme](#).

In der folgenden Liste der threadsicheren Befehle weist ein Stern (*) darauf hin, dass der entsprechend gekennzeichnete Befehl nur unter bestimmten Bedingungen threadsicher ist.

- Solche Befehle für Programmverbindungen, Dateisteuerung, temporären Speicher und transiente Daten sind unter den folgenden Umständen threadsicher:
 - Das Programm, die Datei oder die Warteschlange, auf die sich der Befehl bezieht, ist als lokal definiert.
 - Das Programm, die Datei oder die Warteschlange, auf die sich der Befehl bezieht, ist als fern definiert und der Zugriff auf die Ressource erfolgt durch Distributed Program Link oder durch eine Funktionsverlagerung an eine ferne CICS-Region über eine IPIC-Verbindung.
 - Bei Befehlen für die Dateisteuerung ist die Datei, auf die sich der Befehl bezieht, eine VSAM-RLS-Datei oder eine Coupling-Facility-Datentabelle.
- Unter den folgenden Bedingungen sind diese Befehle nicht threadsicher:
 - Der Zugriff auf die Ressource durch Distributed Program Link oder durch eine Funktionsverlagerung an eine ferne CICS-Region erfolgt über einen anderen Verbindungstyp.
 - Bei Befehlen für die Dateisteuerung ist die Datei, auf die sich der Befehl bezieht, eine gemeinsam genutzte Datentabelle oder eine BDAM-Datei.
- Der Befehl **WRITE OPERATOR** ist threadsicher, kann jedoch nicht aus einem Tasksteuerblock mit Schlüssel 9 verwendet werden, falls die Option **REPLY** angegeben ist. In diesem Fall wechselt CICS für die Verarbeitung des Befehls zum QR-Tasksteuerblock.

Der Aufruf von DL/I unter Verwendung der maßgeblichen Schnittstelle für die Programmiersprache, z. B. die COBOL-Anweisung **CALL CBLTDLI**, ist threadsicher, wenn dies bei IMS Version 12 oder höher verwendet wird.

Liste der threadsicheren Befehle

- **ABEND**
- **ADDRESS**
- **ASKTIME**
- **ASSIGN**
- **BIF DEEDIT**
- **BIF DIGEST**
- **CHANGE PASSWORD**
- **CHANGE PHRASE**
- **CHANGE TASK**
- **CONVERTTIME**
- **DEFINE COUNTER** und **DEFINE DCOUNTER**
- **DELETE ***
- **DELETE CHANNEL**
- **DELETE CONTAINER (CHANNEL)**
- **DELETE COUNTER** und **DELETE DCOUNTER**

- **DELETEQ TD***
- **DELETEQ TS***
- **DEQ** (Dieser Befehl ist threadsicher, wenn er als lokal definiert ist. Er ist nicht threadsicher, wenn er als global definiert ist.)
- **DOCUMENT CREATE**
- **DOCUMENT DELETE**
- **DOCUMENT INSERT**
- **DOCUMENT RETRIEVE**
- **DOCUMENT SET**
- **ENDBR ***
- **ENQ** (Dieser Befehl ist threadsicher, wenn er als lokal definiert ist. Er ist nicht threadsicher, wenn er als global definiert ist.)
- **ENTER TRACENUM**
- **EXEC DLI**
- **EXTRACT CERTIFICATE**
- **EXTRACT TCPIP**
- **EXTRACT WEB**
- **FETCH ANY**
- **FETCH CHILD**
- **FORMATTIME**
- **FREEMAIN**
- **FREEMAIN64**
- **FREE CHILD**
- **GET CONTAINER (CHANNEL)**
- **GET COUNTER** und **GET DCOUNTER**
- **GETMAIN**
- **GETMAIN64**
- **GET64 CONTAINER**
- **HANDLE ABEND**
- **HANDLE AID**
- **HANDLE CONDITION**
- **IGNORE CONDITION**
- **INVOKE APPLICATION**
- **INVOKE SERVICE**
- **INVOKE WEBSERVICE**
- **LINK ***
- **LOAD**
- **MONITOR**
- **MOVE CONTAINER (CHANNEL)**
- **POP HANDLE**
- **PUSH HANDLE**
- **PUT CONTAINER (CHANNEL)**
- **PUT64 CONTAINER**
- **QUERY CHANNEL**

- **QUERY COUNTER** und **QUERY DCOUNTER**
- **QUERY SECURITY**
- **READ ***
- **READNEXT ***
- **READPREV ***
- **READQ TD***
- **READQ TS***
- **RELEASE**
- **REQUEST ENCYRPTPTKT**
- **REQUEST PASSTICKET**
- **RESETBR ***
- **RETURN**
- **REWIND COUNTER** und **REWIND DCOUNTER**
- **REWRITE ***
- **RUN TRANSID**
- **SIGNAL EVENT**
- **SIGNOFF**
- **SIGNON**
- **SIGNON TOKEN**
- **SOAPFAULT ADD**
- **SOAPFAULT CREATE**
- **SOAPFAULT DELETE**
- **STARTBR ***
- **SUSPEND**
- **SYNCPPOINT** (Der Wiederherstellungsmanager verarbeitet diesen Befehl nach Möglichkeit für einen offenen Tasksteuerblock, um den Wechsel von Tasksteuerblöcken zu minimieren.)
- **SYNCPPOINT ROLLBACK** (Der Wiederherstellungsmanager verarbeitet diesen Befehl nach Möglichkeit für einen offenen Tasksteuerblock, um den Wechsel von Tasksteuerblöcken zu minimieren.)
- **TRANSFORM DATATOJSON**
- **TRANSFORM DATATOXML**
- **TRANSFORM JSONTODATA**
- **TRANSFORM XMLTODATA**
- **UNLOCK ***
- **UPDATE COUNTER** und **UPDATE DCOUNTER**
- **VERIFY PASSWORD**
- **VERIFY PHRASE**
- **VERIFY TOKEN**
- **WAIT EXTERNAL**
- **WAIT JOURNALNAME**
- **WAIT JOURNALNUM**
- **WEB CLOSE**
- **WEB CONVERSE**
- **WEB ENDBROWSE FORMFIELD**
- **WEB ENDBROWSE HTTPHEADER**

- WEB ENDBROWSE QUERYPARM
- WEB EXTRACT
- WEB OPEN
- WEB PARSE URL
- WEB READ FORMFIELD
- WEB READ HTTPHEADER
- WEB READNEXT FORMFIELD
- WEB READNEXT HTTPHEADER
- WEB READ QUERYPARM
- WEB READNEXT QUERYPARM
- WEB RECEIVE
- WEB RETRIEVE
- WEB SEND
- WEB STARTBROWSE FORMFIELD
- WEB STARTBROWSE HTTPHEADER
- WEB STARTBROWSE QUERYPARM
- WEB WRITE HTTPHEADER
- WRITE *
- WRITE JOURNALNAME
- WRITE JOURNALNUM
- WRITE OPERATOR*
- WRITEQ TD*
- WRITEQ TS*
- WSACONTEXT BUILD
- WSACONTEXT DELETE
- WSACONTEXT GET
- WSAEPR CREATE
- XCTL

Kapitel 4. Von allen Befehlen verwendete Datenbereiche für CICS-Werte

Dieser Abschnitt enthält die Datenbereiche für CICS-Werte (CICS Value Data Areas - CVDAs) und ihre numerischen Entsprechungen für alle EXEC CICS-Befehle.

CVDAs werden in [Datenbereiche für CICS-Werte \(CVDAs\)](#) beschrieben.

CVDAs und numerische Werte in alphabetischer Reihenfolge

CVDA	Wert
ABEND	900
ACQFAIL	515
ACQUIRED	69
ACQUIRING	71
ACTIVE	181
ACTIVITY	1002
ADD	291
ADDABLE	41
ADDFAIL	519
ADDRESS	859
ADVANCE	265
AINIT	282
ALARM	501
ALL	856
ALLCONN	169
ALLOCATD	81
ALLQUERY	431
ALLVALUES	860
ALTERABLE	52
ALTERNATE	197
ALTPRTCOPY	446
ANALYZER	1126
AND	1005
ANY	158
APLKYBD	391
APLTEXT	393
APPC	124

CVDA	Wert
APPCPARALLEL	374
APPCSINGLE	373
APPEND	1036
APPLICATION	559
APPLID	303
APPLNAME	338
ASACTL	224
ASCII7	616
ASCII8	617
ASRUNTRAN	1209
ASSEMBLER	150
ASSERTED	1104
ASYNCHRONOUS	570
ATI	75
ATOM	1119
ATOMSERVICE	1179
ATTACH	576
ATTENTION	524
ATTLSAWARE	1205
AUDALARM	395
AUTOACTIVE	630
AUTOARCH	262
AUTOAUTH	1095
AUTOCONN	170
AUTOINACTIVE	631
AUTOINIT	284
AUTOINSTALL	1140
AUTOPAGE	80
AUTOREGISTER	1094
AUTOSTART	618
AUTOTERMID	1120
AUXILIARY	247
AUXPAUSE	313
AUXSTART	312
AUXSTOP	314

CVDA	Wert
AVAILABLE	95
BACKOUT	192
BACKTRANS	397
BACKUPNONBWO	800
BASE	10
BASEAPI	1052
BASESPACE	664
BASE64	50
BASICAUTH	1092
BDAM	2
BEGINSESSION	510
BELOW	159
BGAM	63
BINARY	1038
BINIT	283
BIT	1020
BLK	47
BLOCKED	16
BOTH	1100
BRIDGE	935
BROWSABLE	39
BSAM	61
BTAM	62
BUNDLE	1180
BUSY	612
C	149
CACHE	791
CANCEL	526
CANCELLED	624
CANCELLING	1025
CAPTURESPEC	1195
CATEGORY	474
CBE	582
CBER	581
CCE	583

CVDA	Wert
CD	491
CEDF	370
CERTIFCAUTH	1093
CERTUSER	1061
CFTABLE	833
CFE	584
CGROUP	902
CHANNEL	1047
CHAR	1019
CHUNKEND	737
CHUNKNO	735
CHUNKYES	736
CICS	660
CICSAPI	1052
CICSDATAKEY	379
CICSECURITY	195
CICSEXECKEY	381
CICSTABLE	101
CKOPEN	1055
CLASSCACHE	1075
CLEAR	640
CLICONVERT	743
CLIENT	1122
CLIENTAUTH	1032
CLIENTCERT	1087
CLOSE	741
CLOSED	19
CLOSEFAILED	349
CLOSELEAVE	261
CLOSEREQ	22
CLOSING	21
CLOUD	848
CMDPROT	673
CMDSECEXT	207
CMDSECNO	205

CVDA	Wert
CMDSECYES	207
COBOL	151
COBOLII	375
COLD	788
COLDACQ	72
COLDQUERY	433
COLDSTART	266
COLLECTION	473
COLOR	399
COMMAREA	1048
COMMIT	208
COMMITFAIL	792
COMPAT	107
COMPLETE	1026
COMPOSITE	1003
COMPRESS	1130
CONFFREE	82
CONFIG	1232
CONFRECV	83
CONFSEND	84
CONNECT	903
CONNECTED	690
CONNECTING	904
CONNECTION	755
CONSISTENT	723
CONSOLE	66
CONTAINER	869
CONTENTION	836
CONTEXT	574
CONTROLSHUT	623
CONVERSE	600
CONVIDLE	518
COORDINATOR	770
COPID	908
COPY	401

CVDA	Wert
CORBA	1101
CORBASERVER	1141
CPLUSPLUS	148
CREATE	67
CREATESPI	844
CRITICAL	11
CSDAPI	840
CSDBATCH	841
CSIGN	905
CTERM	906
CTLGALL	632
CTLGMODIFY	633
CTLGNONE	634
CTRLABLE	56
CTX	907
CURRENT	260
CURRENTPGM	870
CUSERID	909
CUSTOM	579
DAEOPT	684
DATA	508
DATASET	756
DATASETFULL	793
DATASTREAM	543
DB2CONN	1142
DB2ENTRY	1143
DB2TRAN	1184
DEADLOCK	794
DEBUG	1082
DEC	46
DEFAULT	198
DEFAULTUSER	1070
DEFINITION	990
DEFRESP1	497
DEFRESP1OR2	528

CVDA	Wert
DEFRESP2	498
DEFRESP3	499
DELAY	637
DELETABLE	43
DELETE	292
DELETEFAIL	520
DELEXITERROR	795
DEREGERROR	679
DEREGISTERED	678
DEST	235
DIAGNOSTICS	1231
DISABLED	24
DISABLEDHOST	1125
DISABLING	25
DISCARDFAIL	513
DISCARDING	1118
DISCONNING	910
DISCREQ	444
DISK1	252
DISK2	253
DISK2PAUSE	254
DISPATCHABLE	228
DISPATCHER	1144
DJAR	1189
DOCDELETE	295
DOCTEMPLATE	1145
DOESNOTEQUAL	861
DOESNOTEXIST	872
DOESNOTSTART	862
DORMANT	1024
DPLSUBSET	383
DRAINING	619
DREPAPI	843
DREPBATCH	845
DS3270	615

CVDA	Wert
DSIE	1206
DUALCASE	403
DUMMY	781
DUMP	1223
DUMPCODE	1218
DUPERROR	106
DUPNOREPLACE	105
DUPREPLACE	104
DYNAMIC	178
EB	490
EBCDIC	1039
ECI	1098
EJB	1102
EMERGENCY	268
EMPTY	210
EMPTYREQ	31
ENABLED	23
ENABLING	51
ENDAFFINITY	790
ENQMODEL	1188
ENQUEUE	1146
ENTRYPOINT	993
EPADAPTER	1196
EPADAPTERSET	1200
EQUAL	911
EQUALS	863
ESDS	5
EVENT	334
EVENTBINDING	1191
EVENTPROCESS	1192
EVENTUAL	3
EXCEPT	332
EXCEPTRESP	523
EXCI	650
EXCTL	48

CVDA	Wert
EXECENQ	751
EXECENQADDR	752
EXECINSTALL	846
EXISTS	871
EXIT	912
EXITTRACE	362
EXPECT	747
EXPIRED	1017
EXPORT	991
EXTENDED	1128
EXTENDEDDES	405
EXTRA	221
EXTSECURITY	194
FAILED	782
FAILEDDBKOUT	357
FAILINGBKOUT	358
FAULTTOEPR	853
FCLOSE	273
FEED	472
FILE	238
FINALQ	183
FINPUT	270
FIRED	1001
FIRSTINIT	625
FIRSTQ	182
FIXED	12
FLUSH	783
FMH	502
FMHPARM	385
FOPEN	272
FORCE	342
FORCECANCEL	655
FORCECLOSE	351
FORCED	1013
FORCEPURGE	237

CVDA	Wert
FORCLOSING	353
FORMATEDF	606
FORMATTED	542
FORMFEED	407
FOUTPUT	271
FREE	85
FREEING	94
FROMEPR	850
FULL	212
FULLAPI	384
FULLAUTO	1073
FWDRECOVABLE	354
GATHER	1224
GENERIC	651
GET	748
GMT	604
GOHIGHERTHAN	873
GOINGOUT	172
GOLOWERTHAN	874
GREATERTHAN	864
GROUP	913
GROUPRESYNC	947
GRPLIST	842
GTFSTART	317
GTFSTOP	318
HEAD	750
HEAP	1229
HEURBACKOUT	758
HEURCOMMIT	757
HEX	45
HFORM	409
HFSFILE	1040
HIGH	914
HILIGHT	413
HOLD	163

CVDA	Wert
HOME	1103
HOSTNAME	302
HTTP	1096
HTTPNO	1034
HTTPS	1029
HTTPYES	1033
IDENTIFY	1074
IDNTY	328
IGNORE	1
IGNORERR	816
IIOP	1097
IMMCLOSE	350
IMMCLOSING	352
IMMEDIATE	2
IMMQUIESCED	706
IMPORT	992
INACTIVE	378
INBOUND	547
INCOMPLETE	1014
INDEXRECFULL	796
INDIRECT	122
INDOUBT	620
INFLIGHT	621
INITCOMPLETE	628
INITIAL	789
INITING	1112
INOUT	532
INPUT	226
INSERVICE	73
INSTALLED	550
INSTALLFAIL	512
INTERNAL	1058
INTRA	222
INTSTART	310
INTSTOP	311

CVDA	Wert
INVALID	359
IOERROR	797
IPCONN	1176
IPECI	936
IPIC	1089
IPV4	300
IPV6	301
IRC	121
ISNOTGREATER	865
ISNOTLESS	866
ISOLATE	658
JAVA	147
JAVACORE	1228
JOURNALMODEL	1185
JOURNALNAME	1147
JOURNALNUM	1148
JVM	1080
JVMPPOOL	1149
JVMPROFILE	1150
JVMPROGRAM	1151
JVMSERVER	1193
JWT	1217
KATAKANA	415
KEEP	786
KERBEROS	1105
KEYED	8
KILL	232
KSDS	6
LCKSTRUCFULL	832
LEAVE	811
LESSTHAN	867
LE370	377
LIBERTY	1222
LIBRARY	1177
LIC	493

CVDA	Wert
LIGHTPEN	417
LINK	577
LOAD	834
LOC24	159
LOC31	158
LOCAL	605
LOCKING	837
LOG	54
LOGICAL	216
LOGREPLICATE	1198
LOGTERM	269
LOSE	544
LOW	915
LPA	165
LSRPOOL	1152
LUP	541
LUSTAT	525
LUW	246
LU61	125
MAIN	248
MAP	155
MAPSET	155
MCHCTL	241
MDT	506
METADATA	857
MESSAGEID	1197
MIXED	403
MNPS	999
MOD	813
MODEANY	189
MODEL	370
MODE24	190
MODE31	191
MONITOR	1153
MORE	492

CVDA	Wert
MQCONN	1175
MRO	938
MSRCONTROL	419
MTOM	460
MTOMNOXOP	469
MVS	780
MVSTCB	1154
NEGATIVE	530
NEW	28
NEWCOPY	167
NEWSESSION	485
NOALARM	500
NOALTPRTCOPY	447
NOANALYZER	1127
NOAPLKYBD	392
NOAPLTEXT	394
NOAPPEND	1037
NOAPPLNAME	339
NOATI	76
NOAUDALARM	396
NOAUTHENTIC	1091
NOAUTO	1071
NOAUTOARCH	263
NOAUTOSTART	1208
NOBACKOUT	193
NOBACKTRANS	398
NOCEDF	371
NOCLASSCACHE	1076
NOCLEAR	641
NOCLICONVERT	744
NOCLOSE	742
NOCMDPROT	674
NOCOLOR	400
NOCOMPAT	108
NOCOMPRESS	1131

CVDA	Wert
NOCONNECT	916
NOCONV	556
NOCONVERSE	601
NOCONVERT	734
NOCOPY	402
NOCREATE	68
NOCTL	223
NODAEOPT	685
NODE	1155
NODEBUG	1083
NODEJSAPP	1215
NODISCREQ	445
NODOCDELETE	296
NODUALCASE	404
NOEMPTYREQ	32
NOEVENT	335
NOEXCEPT	333
NOEXCTL	49
NOEXIT	917
NOEXITTRACE	363
NOEXTENDEDDES	406
NOFMH	503
NOFMHPARM	386
NOFORCE	1054
NOFORMATEDF	607
NOFORMFEED	408
NOHFORM	410
NOHILIGHT	414
NOHOLD	164
NOIDENTY	329
NOINCONVERT	745
NOISOLATE	657
NOJVM	1081
NOKATAKANA	416
NOKEEP	787

CVDA	Wert
NOLIGHTPEN	418
NOLOAD	835
NOLOG	55
NOLOGREPLICA	1199
NOLOSTLOCKS	710
NOMDT	507
NOMSGJRNL	531
NOMSRCONTROL	420
NOMTOM	461
NOMTOMNOXOP	470
NONAUTOCONN	171
NONCICS	661
NONCRITICAL	1090
NONE	496
NONLE370	1084
NONRESIDENT	1202
NONTRANS	572
NOOBFORMAT	422
NOOBOPERID	388
NOOUTCONVERT	746
NOOUTLINE	424
NOPARTITIONS	426
NOPERF	331
NOPHASEOUT	1110
NOPRESETSEC	243
NOPRINTADAPT	428
NOPROGSYMBOL	430
NOPRTCOPY	449
NOPS	997
NOQUERY	432
NORECOVDATA	700
NOREENTPROT	681
NORELEASE	918
NORELREQ	443
NOREPLICATOR	1212

CVDA	Wert
NORESCE	337
NORESYNC	934
NORETAINED	711
NOREUSE	294
NORMAL	1016
NORMALBKOUT	356
NORMALRESP	522
NORMI	775
NOROLLBACK	919
NOSECURITY	196
NOSENDMTOM	464
NOSHUTDOWN	289
NOSOSI	435
NOSPI	694
NOSRVCONVERT	739
NOSSL	1031
NOSTSN	487
NOSWITCH	285
NOSYNCPPOINT	603
NOSYSDUMP	185
NOSYSLOG	784
NOTADDABLE	42
NOTALLOWED	327
NOTALTERABLE	53
NOTAPPLIC	1
NOTASKSTART	608
NOTBRWBLE	40
NOTBUSY	613
NOTCDEB	495
NOTCONNECTED	691
NOTCTRLABLE	57
NOTDEFINED	659
NOTDELBLE	44
NOTDYNAMIC	1021
NOTEMPTY	211

CVDA	Wert
NOTEXTENDED	1129
NOTERMINAL	214
NOTEXTKYBD	437
NOTEXTPRINT	439
NOTFIRED	1000
NOTFWDRCVBLE	361
NOTHOTPOOL	1066
NOTINBOUND	546
NOTINIT	376
NOTINSTALLED	551
NOTKEYED	9
NOTLPA	166
NOTPENDING	127
NOTPURGEABLE	161
NOTRANDUMP	187
NOTREADABLE	36
NOTREADY	259
NOTRECOVABLE	30
NOTREQUIRED	667
NOTRLS	721
NOTROUTABLE	1022
NOTSOS	669
NOTSUPPORTED	15
NOTSUSPENDED	1027
NOTTABLE	100
NOTTI	78
NOTUPDBLE	38
NOTWAIT	920
NOUCTRAN	451
NOUSER	1011
NOVALIDATION	441
NOVFORM	412
NOWAIT	341
NOWRITE	275
NOXOPDIRECT	466

CVDA	Wert
NOXOPSUPPORT	468
NOZCPTRACE	365
NRS	774
N906	931
N906D	921
OBFORMAT	421
OBOPERID	387
OBTAINING	96
OFF	200
OK	274
OLD	26
OLDCOPY	162
OLDSESSION	486
ON	201
OPEN	18
OPENAPI	1053
OPENERERROR	798
OPENING	20
OPENINPUT	256
OPENOUTPUT	257
OPID	933
OPTIONAL	326
OPTIONS	759
OR	1006
OSGI	1221
OTS	1107
OUTLINE	423
OUTPUT	227
OUTSERVICE	74
OWNER	753
PAGEABLE	79
PARTITIONS	425
PARTITIONSET	156
PARTNER	1182
PATCH	1220

CVDA	Wert
PATH	11
PDSMEMBER	1035
PENDBEGIN	558
PENDDATA	560
PENDFREE	86
PENDING	126
PENDINIT	1113
PENDPASS	565
PENDRECEIVE	87
PENDRELEASE	562
PENDRESOLVE	1114
PENDSTART	561
PENDSTSN	557
PENDUNSOL	564
PERF	330
PERMANENT	484
PHASEIN	168
PHASEOUT	1111
PHYSICAL	215
PIPELINE	1124
PLI	152
PL1	152
POLICYSCOPE	994
POOL	922
POSITIVE	529
POST	636
POSTCOMMAND	697
PRECOMMAND	696
PRESETSEC	242
PRIMARY	110
PRINCIPAL	199
PRINTADAPT	427
PRIVATE	174
PROCESS	1010
PROCESSTYPE	1190

CVDA	Wert
PROFILE	157
PROGAUTO	1072
PROGRAM	154
PROGRAMDEF	1178
PROGRAMINIT	698
PROGSYMBOL	429
PROTECTED	504
PROVIDER	1041
PRTCOPY	448
PUBLISH	1085
PURGE	236
PURGEABLE	160
PUT	749
QR	1057
QUASIRENT	1050
QUEUE	814
QUIESCED	707
QUIESCING	708
READABLE	35
READBACK	209
READONLY	275
READY	258
REBUILD	923
RECEIVE	88
RECEIVER	1046
RECONNECT	924
RECOVDATA	701
RECOVERABLE	29
RECOVERED	277
RECOVERLOCKS	712
RECOVERY	1156
REENTPROT	680
REFPARMS	858
REFRESH	1227
REFRESHPKGS	1225

CVDA	Wert
REGERROR	677
REGION	575
REGISTERED	670
REJECT	815
RELATED	675
RELEASE	563
RELEASED	70
RELEASING	549
RELOAD	1078
RELOADING	1079
RELREQ	442
REMLOSTLOCKS	713
REMOTE	4
REMOVE	276
REMSESSION	740
REMTABLE	103
REPEATABLE	724
REPLICATOR	1211
REPLYTOEPR	852
REQCONTEXT	854
REQUEST	249
REQUESTER	1042
REQUESTMODEL	1157
REQUIRED	666
REREAD	812
RESET	290
RESETLOCKS	714
RESIDENT	1201
RESOLVED	347
RESOLVING	1115
RESRCE	336
RESSECEXT	204
RESSECINT	203
RESSECNO	202
RESSECYES	204

CVDA	Wert
RESPCONTEXT	855
RESPECTED	1109
RESYNC	702
RETAINED	715
RETRACT	1086
RETRY	716
REUSE	293
REVERTED	264
RFC1123	646
RFC3339	647
RLS	720
RLSACTIVE	730
RLSGONE	799
RLSINACTIVE	731
RLSSERVER	761
RMI	771
ROLLBACK	89
ROUTABLE	1023
ROUTE	638
RPG	153
RRCOMMITFAIL	830
RRDS	7
RREPL	1210
RRINDOUBT	831
RRMS	1063
RRSUR	939
RTR	527
RU	494
RUNNING	229
RZINSTOR	940
SAMESENDMTOM	462
SCAN	1088
SCHEDULER	941
SCS	614
SECERROR	1214

CVDA	Wert
SECONDINIT	626
SECURITY	1216
SECUSER	1060
SEND	90
SENDER	1045
SENDMTOM	463
SERVER	1123
SERVERDUMP	1226
SERVICE	475
SESSION	372
SESSIONFAIL	517
SESSIONLOST	516
SESSIONS	1187
SETFAIL	514
SHARE	27
SHARED	173
SHUNTED	762
SHUTDISABLED	645
SHUTDOWN	288
SHUTENABLED	644
SIGN	925
SIGNEDOFF	245
SIGNEDON	244
SINGLEOFF	324
SINGLEON	323
SKIP	810
SMF	255
SNAPTRACE	1230
SNPS	998
SOAP	1044
SOCKET	942
SOMEAVAIL	1204
SOS	668
SOSABOVE	683
SOSBELOW	682

CVDA	Wert
SOSI	434
SPECIAL	98
SPECIFIC	652
SPECTRACE	177
SPI	693
SPRSTRACE	175
SQLCODE	926
SRVCONVERT	738
SSL	1030
STANDARD	97
STANDBY	629
STANTRACE	176
START	635
STARTED	609
STARTING	772
STARTTERM	943
STARTING	772
STARTSWITH	868
STARTUP	180
STATIC	179
STATS	1158
STOPPED	610
STOPPING	348
STORAGE	1159
STREAMNAME	1160
STSN	509
STSNSET	488
STSNTEST	489
SUBORDINATE	773
SUBPOOL	1161
SUBSPACE	663
SUMMUNLIKE	566
SUPPORTED	1106
SURROGATE	371
SUSPENDED	231

CVDA	Wert
SWITCH	188
SWITCHALL	287
SWITCHING	225
SWITCHNEXT	286
SYNCFREE	91
SYNCHRONOUS	571
SYNCPOINT	602
SYNCRECV	92
SYNCSEND	93
SYSCONNECT	653
SYSDUMP	184
SYSDUMPCODE	1162
SYSLOG	785
SYSPLEX	656
SYSTEM	643
SYSTEMOFF	320
SYSTEMON	319
TABLE	847
TABLEMGR	1163
TABLEONLY	1219
TAKEOVER	111
TAPE1	250
TAPE2	251
TASK	233
TASKSUBPOOL	1164
TASKSTART	611
TCAM	64
TCAMSNA	65
TCEXITALL	366
TCEXITALLOFF	369
TCEXITNONE	368
TCEXITSYSTEM	367
TCLASS	239
TCPIP	802
TCPIPSERVICE	1166

CVDA	Wert
TDQ	767
TDQUEUE	767
TEMPLATENAME	1168
TEMPORARY	483
TERM	234
TERMINAL	213
TERMINATE	1077
TEXTKYBD	436
TEXTPRINT	438
THIRDINIT	627
THREADSAFE	1051
TIME	240
TIMEOUT	511
TIMER	1004
TOEPR	851
TPOOL	932
TPS55M2	552
TPS55M3	553
TPS55M4	554
TPS55M5	555
TRACE	760
TRANCLASS	1169
TRANDATA	944
TRANDUMP	186
TRANDUMPCODE	1170
TRANIDONLY	452
TRANS	573
TRANSACTION	1171
TRANSTART	580
TSMODEL	1183
TSQ	768
TSQUEUE	768
TTI	77
TWAIT	927
TX	929

CVDA	Wert
TXID	928
TYPETERM	1186
T3278M2	533
T3278M3	534
T3278M4	535
T3278M5	536
T3279M2	537
T3279M3	538
T3279M4	539
T3279M5	540
UCTRAN	450
UKOPEN	1056
UNATTEMPTED	820
UNAVAILABLE	672
UNBLOCKED	17
UNCOMMITTED	722
UNCONNECTED	703
UNDEFINED	14
UNDETERMINED	355
UNENABLED	33
UNENABLING	34
UNEXPECTED	1015
UNEXPIRED	1018
UNINSTALLED	548
UNKNOWN	774
UNPROTECTED	505
UNQUIESCED	709
UNREGISTERED	671
UNRESOLVED	1116
UNSOLDATA	521
UNUSABLE	1117
UOW	246
UPDATABLE	37
UPDATING	1049
URIMAP	1173

CVDA	Wert
URMTERID	1121
USER	642
USERDATAKEY	380
USEREXECKEY	382
USERID	930
USEROFF	322
USERON	321
USERTABLE	102
UTC	1203
VALID	360
VALIDATION	440
VARIABLE	13
VERIFY	1099
VFORM	411
VRRDS	732
VSAM	3
VTAM	60
WAIT	340
WAITCOMMIT	763
WAITER	754
WAITFORGET	622
WAITING	765
WAITRMI	766
WAITRRMS	1063
WARMSTART	267
WBE	585
WEB	945
WEBSERVICE	1174
WIN	545
WMQ	578
XCF	665
XM	123
XMLTRANSFORM	1194
XMRUNTRAN	946
XNOTDONE	144

CVDA	Wert
XOK	143
XOPDIRECT	465
XOPSUPPORT	467
XPLINK	1068
ZCPTRACE	364

Anmerkung: VTAM ist jetzt z/OS Communications Server.

CVDA-Werte für die Option DEVICE

CVDA-Reihenfolge	
BATCHLU	191
BIPROG	160
BISYNCH	128
CDRDLPR	24
CONTNLU	189
HARDCOPY	32
INTACTLU	190
ISCMCONV	209
LUCMODGRP	210
LUCSESS	211
LUTYPE4	193
LUTYPE6	192
MAGTAPE	20
RESSYS	208
SDLC	176
SEQDISK	18
SYSTEM3	161
SYSTEM7	2
SYS370	164
SYS7BSCA	166
TCONSOLE	8
TELETYPE	34
TTCAM	80
TWX3335	33
T1050	36
T1053	74

CVDA-Reihenfolge	
T2260L	65
T2260R	72
T2265	76
T2740	40
T2741BCD	43
T2741COR	42
T2770	130
T2780	132
T2980	134
T3275R	146
T3277L	153
T3277R	145
T3284L	155
T3284R	147
T3286L	156
T3286R	148
T3600BI	138
T3601	177
T3614	178
T3650ATT	186
T3650PIPE	184
T3650USER	187
T3653HOST	185
T3735	136
T3740	137
T3780	133
T3790	180
T3790SCSP	182
T3790UP	181
T7770	1
VIDEOTERM	64

Numerische Reihenfolge	
1	T7770
2	SYSTEM7
8	TCONSOLE

Numerische Reihenfolge	
18	SEQDISK
20	MAGTAPE
24	CDRDLPRT
32	HARDCOPY
33	TWX3335
34	TELETYPE
36	T1050
40	T2740
42	T2741COR
43	T2741BCD
64	VIDEOTERM
65	T2260L
72	T2260R
74	T1053
76	T2265
80	TTCAM
128	BISYNCH
130	T2770
132	T2780
133	T3780
134	T2980
136	T3735
137	T3740
138	T3600BI
145	T3277R
146	T3275R
147	T3284R
148	T3286R
153	T3277L
155	T3284L
156	T3286L
160	BIPROG
161	SYSTEM3
164	SYS370
166	SYS7BSCA

Numerische Reihenfolge	
176	SDLC
177	T3601
178	T3614
180	T3790
181	T3790UP
182	T3790SCSP
184	T3650PIPE
185	T3653HOST
186	T3650ATT
187	T3650USER
189	CONTNLU
190	INTACTLU
191	BATCHLU
192	LUTYPE6
193	LUTYPE4
208	RESSYS
209	ISCMMCONV
210	LUCMODGRP
211	LUCSESS

CVDAs und numerische Werte in numerischer Reihenfolge

Wert	CVDA
1	IGNORE
1	NOTAPPLIC
2	BDAM
2	IMMEDIATE
3	EVENTUAL
3	VSAM
4	REMOTE
5	ESDS
6	KSDS
7	RRDS
8	KEYED
9	NOTKEYED
10	BASE

Wert	CVDA
11	CRITICAL
11	PATH
12	FIXED
13	VARIABLE
14	UNDEFINED
15	NOTSUPPORTED
16	BLOCKED
17	UNBLOCKED
18	OPEN
19	CLOSED
20	OPENING
21	CLOSING
22	CLOSEREQUEST
23	ENABLED
24	DISABLED
25	DISABLING
26	OLD
27	SHARE
28	NEW
29	RECOVERABLE
30	NOTRECOVABLE
31	EMPTYREQ
32	NOEMPTYREQ
33	UNENABLED
34	UNENABLING
35	READABLE
36	NOTREADABLE
37	UPDATABLE
38	NOTUPDATABLE
39	BROWSABLE
40	NOTBROWSABLE
41	ADDABLE
42	NOTADDABLE
43	DELETABLE
44	NOTDELETABLE

Wert	CVDA
45	HEX
46	DEC
47	BLK
48	EXCTL
49	NOEXCTL
50	BASE64
51	ENABLING
52	ALTERABLE
53	NOTALTERABLE
54	LOG
55	NOLOG
56	CTRLABLE
57	NOTCTRLABLE
60	VTAM
61	BSAM
62	BTAM
63	BGAM
64	TCAM
65	TCAMSNA
66	CONSOLE
67	CREATE
68	NOCREATE
69	ACQUIRED
70	RELEASED
71	ACQUIRING
72	COLDACQ
73	INSERVICE
74	OUTSERVICE
75	ATI
76	NOATI
77	TTI
78	NOTTI
79	PAGEABLE
80	AUTOPAGEABLE
81	ALLOCATED

Wert	CVDA
82	CONFFREE
83	CONFRECEIVE
84	CONFSEND
85	FREE
86	PENDFREE
87	PENDRECEIVE
88	RECEIVE
89	ROLLBACK
90	SEND
91	SYNCFREE
92	SYNCRECEIVE
93	SYNCSEND
94	FREEING
95	AVAILABLE
96	OBTAINING
97	STANDARD
98	SPECIAL
100	NOTTABLE
101	CICSTABLE
102	USERTABLE
103	REMTABLE
110	PRIMARY
111	TAKEOVER
121	IRC
122	INDIRECT
123	XM
124	APPC
125	LU61
126	PENDING
127	NOTPENDING
143	XOK
144	XNOTDONE
147	JAVA
148	CPLUSPLUS
149	C

Wert	CVDA
150	ASSEMBLER
151	COBOL
152	PLI
152	PL1
153	RPG
154	PROGRAM
155	MAP
155	MAPSET
156	PARTITIONSET
157	PROFILE
158	ANY
158	LOC31
159	BELOW
159	LOC24
160	PURGEABLE
161	NOTPURGEABLE
162	OLDCOPY
163	HOLD
164	NOHOLD
165	LPA
166	NOTLPA
167	NEWCOPY
168	PHASEIN
169	ALLCONN
170	AUTOCONN
171	NONAUTOCONN
172	GOINGOUT
173	SHARED
174	PRIVATE
175	SPRSTRACE
176	STANTRACE
177	SPECTRACE
178	DYNAMIC
179	STATIC
180	STARTUP

Wert	CVDA
181	ACTIVE
182	FIRSTQUIESCE
183	FINALQUIESCE
184	SYSDUMP
185	NOSYSDUMP
186	TRANDUMP
187	NOTRANDUMP
188	SWITCH
189	MODEANY
190	MODE24
191	MODE31
192	BACKOUT
193	NOBACKOUT
194	EXTSECURITY
195	CICSECURITY
196	NOSECURITY
197	ALTERNATE
198	DEFAULT
199	PRINCIPAL
200	OFF
201	ON
202	RESSECNO
203	RESSECINT
204	RESSECEXT
204	RESSECYES
205	CMDSECNO
207	CMDSECEXT
207	CMDSECYES
208	COMMIT
209	READBACK
210	EMPTY
211	NOTEMPTY
212	FULL
213	TERMINAL
214	NOTERMINAL

Wert	CVDA
215	PHYSICAL
216	LOGICAL
221	EXTRA
222	INTRA
223	NOCTL
224	ASACTL
225	SWITCHING
226	INPUT
227	OUTPUT
228	DISPATCHABLE
229	RUNNING
231	SUSPENDED
232	KILL
233	TASK
234	TERM
235	DEST
236	PURGE
237	FORCEPURGE
238	FILE
239	TCLASS
240	TIME
241	MCHCTL
242	PRESETSEC
243	NOPRESETSEC
244	SIGNEDON
245	SIGNEDOFF
246	LUW
247	AUXILIARY
248	MAIN
249	REQUEST
250	TAPE1
251	TAPE2
252	DISK1
253	DISK2
254	DISK2PAUSE

Wert	CVDA
255	SMF
256	OPENINPUT
257	OPENOUTPUT
258	READY
259	NOTREADY
260	CURRENT
261	CLOSELEAVE
262	AUTOARCH
263	NOAUTOARCH
264	REVERTED
265	ADVANCE
266	COLDSTART
267	WARMSTART
268	EMERGENCY
269	LOGTERM
270	FINPUT
271	FOUTPUT
272	FOPEN
273	FCLOSE
274	OK
275	NOWRITE
275	READONLY
276	REMOVE
277	RECOVERED
282	AINIT
283	BINIT
284	AUTOINIT
285	NOSWITCH
286	SWITCHNEXT
287	SWITCHALL
288	SHUTDOWN
289	NOSHUTDOWN
290	RESET
291	ADD
292	DELETE

Wert	CVDA
293	REUSE
294	NOREUSE
300	IPV4
301	IPV6
302	HOSTNAME
303	APPLID
310	INTSTART
311	INTSTOP
312	AUXSTART
313	AUXPAUSE
314	AUXSTOP
317	GTFSTART
318	GTFSTOP
319	SYSTEMON
320	SYSTEMOFF
321	USERON
322	USEROFF
323	SINGLEON
324	SINGLEOFF
326	OPTIONAL
327	NOTALLOWED
328	IDNTY
329	NOIDNTY
330	PERF
331	NOPERF
332	EXCEPT
333	NOEXCEPT
334	EVENT
335	NOEVENT
336	RESRCE
337	NORESACE
338	APPLNAME
339	NOAPPLNAME
340	WAIT
341	NOWAIT

Wert	CVDA
347	RESOLVED
348	STOPPING
349	CLOSEFAILED
350	IMMCLOSE
351	FORCECLOSE
352	IMMCLOSING
353	FORCECLOSING
354	FWDRECOVABLE
355	UNDETERMINED
356	NORMALBKOUT
357	FAILEDDBKOUT
358	FAILINGBKOUT
359	INVALID
360	VALID
361	NOTFWDRCVBLE
362	EXITTRACE
363	NOEXITTRACE
364	ZCPTRACE
365	NOZCPTRACE
366	TCEXITALL
367	TCEXITSYSTEM
368	TCEXITNONE
369	TCEXITALLOFF
370	CEDF
370	MODEL
371	NOCEDF
371	SURROGATE
372	SESSION
373	APPCSINGLE
374	APPCPARALLEL
375	COBOLII
376	NOTINIT
377	LE370
378	INACTIVE
379	CICSATAKEY

Wert	CVDA
380	USERDATAKEY
381	CICSEXECKEY
382	USEREXECKEY
383	DPLSUBSET
384	FULLAPI
385	FMHPARM
386	NOFMHPARM
387	OBOPERID
388	NOOBOPERID
391	APLKYBD
392	NOAPLKYBD
393	APLTEXT
394	NOAPLTEXT
395	AUDALARM
396	NOAUDALARM
397	BACKTRANS
398	NOBACKTRANS
399	COLOR
400	NOCOLOR
401	COPY
402	NOCOPY
403	MIXED
404	NODUALCASE
405	EXTENDEDDES
406	NOEXTENDEDDES
407	FORMFEED
408	NOFORMFEED
409	HFORM
410	NOHFORM
411	VFORM
412	NOVFORM
413	HILIGHT
414	NOHILIGHT
415	KATAKANA
416	NOKATAKANA

Wert	CVDA
417	LIGHTPEN
418	NOLIGHTPEN
419	MSRCONTROL
420	NOMSRCONTROL
421	OBFORMAT
422	NOOBFORMAT
423	OUTLINE
424	NOOUTLINE
425	PARTITIONS
426	NOPARTITIONS
427	PRINTADAPT
428	NOPRINTADAPT
429	PROGSYMBOL
430	NOPROGSYMBOL
431	ALLQUERY
432	NOQUERY
433	COLDQUERY
434	SOSI
435	NOSOSI
436	TEXTKYBD
437	NOTEXTKYBD
438	TEXTPRINT
439	NOTEXTPRINT
440	VALIDATION
441	NOVALIDATION
442	RELREQ
443	NORELREQ
444	DISCREQ
445	NODISCREQ
446	ALTPRTCOPY
447	NOALTPRTCOPY
448	PRTCOPY
449	NOPRTCOPY
450	UCTRAN
451	NOUCTRAN

Wert	CVDA
452	TRANIDONLY
460	MTOM
461	NOMTOM
462	SAMESENDMTOM
463	SENDMTOM
464	NOSENDMTOM
465	XOPDIRECT
466	NOXOPDIRECT
467	XOPSUPPORT
468	NOXOPSUPPORT
469	MTOMNOXOP
470	NOMTOMNOXOP
472	FEED
473	COLLECTION
474	CATEGORY
475	SERVICE
483	TEMPORARY
484	PERMANENT
485	NEWSESSION
486	OLDSESSION
487	NOSTSN
488	STSNSET
489	STSNTEST
490	EB
491	CD
492	MORE
493	LIC
494	RU
495	NOTCDEB
496	NONE
497	DEFRESP1
498	DEFRESP2
499	DEFRESP3
500	NOALARM
501	ALARM

Wert	CVDA
502	FMH
503	NOFMH
504	PROTECTED
505	UNPROTECTED
506	MDT
507	NOMDT
508	DATA
509	STSN
510	BEGINSESSION
511	TIMEOUT
512	INSTALLFAIL
513	DISCARDFAIL
514	SETFAIL
515	ACQFAIL
516	SESSIONLOST
517	SESSIONFAIL
518	CONVIDLE
519	ADDFAIL
520	DELETEFAIL
521	UNSOLDATA
522	NORMALRESP
523	EXCEPTRESP
524	ATTENTION
525	LUSTAT
526	CANCEL
527	RTR
528	DEFRESP1OR2
529	POSITIVE
530	NEGATIVE
531	NOMSGJRNL
532	INOUT
533	T3278M2
534	T3278M3
535	T3278M4
536	T3278M5

Wert	CVDA
537	T3279M2
538	T3279M3
539	T3279M4
540	T3279M5
541	LUP
542	FORMATTED
543	DATASTREAM
544	LOSE
545	WIN
546	NOTINBOUND
547	INBOUND
548	UNINSTALLED
549	RELEASING
550	INSTALLED
551	NOTINSTALLED
552	TPS55M2
553	TPS55M3
554	TPS55M4
555	TPS55M5
556	NOCONV
557	PENDSTSN
558	PENDBEGIN
559	APPLICATION
560	PENDDATA
561	PENDSTART
562	PENDRELEASE
563	RELEASE
564	PENDUNSOL
565	PENDPASS
566	SUMMUNLIKE
570	ASYNCHRONOUS
571	SYNCHRONOUS
572	NONTRANS
573	TRANS
574	CONTEXT

Wert	CVDA
575	REGION
576	ATTACH
577	LINK
578	WMQ
579	CUSTOM
580	TRANSTART
581	CBER
582	CBE
583	CCE
584	CFE
585	WBE
600	CONVERSE
601	NOCONVERSE
602	SYNCPOINT
603	NOSYNCPOINT
604	GMT
605	LOCAL
606	FORMATEDF
607	NOFORMATEDF
608	NOTASKSTART
609	STARTED
610	STOPPED
611	TASKSTART
612	BUSY
613	NOTBUSY
614	SCS
615	DS3270
616	ASCII7
617	ASCII8
618	AUTOSTART
619	DRAINING
620	INDOUBT
621	INFLIGHT
622	WAITFORGET
623	CONTROLSHUT

Wert	CVDA
624	CANCELLED
625	FIRSTINIT
626	SECONDINIT
627	THIRDINIT
628	INITCOMPLETE
629	STANDBY
630	AUTOACTIVE
631	AUTOINACTIVE
632	CTLGALL
633	CTLGMODIFY
634	CTLGNONE
635	START
636	POST
637	DELAY
638	ROUTE
640	CLEAR
641	NOCLEAR
642	USER
643	SYSTEM
644	SHUTENABLED
645	SHUTDISABLED
646	RFC1123
647	RFC3339
650	EXCI
651	GENERIC
652	SPECIFIC
655	FORCECANCEL
656	SYSPLEX
657	NOISOLATE
658	ISOLATE
659	NOTDEFINED
660	CICS
661	NONCICS
663	SUBSPACE
664	BASESPACE

Wert	CVDA
665	XCF
666	REQUIRED
667	NOTREQUIRED
668	SOS
669	NOTSOS
670	REGISTERED
671	UNREGISTERED
672	UNAVAILABLE
673	CMDPROT
674	NOCMDPROT
675	RELATED
677	REGERROR
678	DEREGISTERED
679	DEREGERROR
680	REENTPROT
681	NOREENTPROT
682	SOSBELOW
683	SOSABOVE
684	DAE
685	NODAE
690	CONNECTED
691	NOTCONNECTED
693	SPI
694	NOSPI
696	PRECOMMAND
697	POSTCOMMAND
698	PROGRAMINIT
700	NORECOVDATA
701	RECOVDATA
702	RESYNC
703	UNCONNECTED
706	IMMQUESCED
707	QUIESCED
708	QUIESCING
709	UNQUIESCED

Wert	CVDA
710	NOLOSTLOCKS
711	NORETAINED
712	RECOVERLOCKS
713	REMLOSTLOCKS
714	RESETLOCKS
715	RETAINED
716	RETRY
720	RLS
721	NOTRLS
722	UNCOMMITTED
723	CONSISTENT
724	REPEATABLE
730	RLSACTIVE
731	RLSINACTIVE
732	VRRDS
734	NOCONVERT
735	CHUNKNO
736	CHUNKYES
737	CHUNKEND
738	SRVCONVERT
739	NOSRVCONVERT
740	REMSESSION
741	CLOSE
742	NOCLOSE
743	CLICONVERT
744	NOCLICONVERT
745	NOINCONVERT
746	NOOUTCONVERT
747	EXPECT
748	GET
749	PUT
750	HEAD
751	EXECENQ
752	EXECENQADDR
753	OWNER

Wert	CVDA
754	WAITER
755	CONNECTION
756	DATASET
757	HEURCOMMIT
758	HEURBACKOUT
759	OPTIONS
760	TRACE
761	RLSSERVER
762	SHUNTED
763	WAITCOMMIT
765	WAITING
766	WAITRMI
767	TDQ (TDQUEUE)
768	TSQ (TSQUEUE)
770	COORDINATOR
771	RMI
772	STARTING
773	SUBORDINATE
774	NRS
774	UNKNOWN
775	NORMI
780	MVS
781	DUMMY
782	FAILED
783	FLUSH
784	NOSYSLOG
785	SYSLOG
786	KEEP
788	COLD
789	INITIAL
790	ENDAFFINITY
791	CACHE
792	COMMITFAIL
793	DATASETFULL
794	DEADLOCK

Wert	CVDA
795	DELEXITERROR
796	INDEXRECFULL
797	IOERROR
798	OPENERERROR
799	RLSGONE
800	BACKUPNONBWO
802	TCPIP
810	SKIP
811	LEAVE
812	REREAD
813	MOD
814	QUEUE
815	REJECT
816	IGNORERR
820	UNATTEMPTED
830	RRCOMMITFAIL
831	RRINDOUBT
832	LCKSTRUCFULL
833	CFTABLE
834	LOAD
835	NOLOAD
836	CONTENTION
837	LOCKING
840	CSDAPI
841	CSDBATCH
842	GRPLIST
843	DREPAPI
844	CREATESPI
845	DREPBATCH
847	TABLE
848	CLOUD
850	FROMEPR
851	TOEPR
852	REPLYTOEPR
853	FAULTTOEPR

Wert	CVDA
854	REQCONTEXT
856	ALL
857	METADATA
858	REFPARMS
859	ADDRESS
860	ALLVALUES
861	DOESNOTEQUAL
862	DOESNOTSTART
863	EQUALS
864	GREATERTHAN
865	ISNOTGREATER
866	ISNOTLESS
867	LESSTHAN
868	STARTSWITH
869	CONTAINER
870	CURRENTPGM
871	EXISTS
872	DOESNOTEXIST
873	GOHIGHERTHAN
874	GOLOWERTHAN
900	ABEND
902	CGROUP
903	CONNECT
904	CONNECTING
905	CSIGN
906	CTERM
907	CTX
908	COPID
909	CUSERID
910	DISCONNING
911	EQUAL
912	EXIT
913	GROUP
914	HIGH
915	LOW

Wert	CVDA
916	NOCONNECT
917	NOEXIT
918	NORELEASE
919	NOROLLBACK
920	NOTWAIT
921	N906D
922	POOL
923	REBUILD
924	RECONNECT
925	SIGN
926	SQLCODE
927	TWAIT
928	TXID
929	TX
930	USERID
931	N906
932	TPOOL
933	OPID
934	NORES SYNC
935	BRIDGE
938	MRO
939	RRSUR
940	RZINSTOR
941	SCHEDULER
942	SOCKET
943	STARTTERM
944	TRAN DATA
945	WEB
946	XMRUNTRAN
947	GROUPRES SYNC
990	DEFINITION
991	EXPORT
992	IMPORT
993	ENTRYPOINT
994	POLICYSCOPE

Wert	CVDA
997	NOPS
998	SNPS
999	MNPS
1000	NOTFIRED
1001	FIRED
1002	ACTIVITY
1003	COMPOSITE
1004	TIMER
1005	AND
1006	OR
1010	PROCESS
1011	NOUSER
1013	FORCED
1014	INCOMPLETE
1015	UNEXPECTED
1016	NORMAL
1017	EXPIRED
1018	UNEXPIRED
1019	CHAR
1020	BIT
1021	NOTDYNAMIC
1022	NOTROUTABLE
1023	ROUTABLE
1024	DORMANT
1025	CANCELLING
1026	COMPLETE
1027	NOTSUSPENDED
1029	HTTPS
1030	SSL
1031	NOSSL
1032	CLIENTAUTH
1033	HTTPYES
1034	HTTPNO
1035	PDSMEMBER
1036	APPEND

Wert	CVDA
1037	NOAPPEND
1038	BINARY
1039	EBCDIC
1040	HFSFILE
1041	PROVIDER
1042	REQUESTER
1044	SOAP
1045	SENDER
1046	RECEIVER
1047	CHANNEL
1048	COMMAREA
1049	UPDATING
1050	QUASIRENT
1051	THREADSAFE
1052	BASEAPI
1052	CICSAPI
1053	OPENAPI
1054	NOFORCE
1055	CKOPEN
1056	UKOPEN
1057	QR
1058	INTERNAL
1060	SECUSER
1061	CERTUSER
1063	WAITRRMS
1066	NOTHOTPOOL
1068	XPLINK
1070	DEFAULTUSER
1071	NOAUTO
1072	PROGAUTO
1073	FULLAUTO
1074	IDENTIFY
1075	CLASSCACHE
1076	NOCLASSCACHE
1077	TERMINATE

Wert	CVDA
1078	RELOAD
1079	RELOADING
1080	JVM
1081	NOJVM
1082	DEBUG
1083	NODEBUG
1084	NONLE370
1085	PUBLISH
1086	RETRACT
1087	CLIENTCERT
1088	SCAN
1089	IPIC
1090	NONCRITICAL
1091	NOAUTHENTIC
1092	BASICAUTH
1093	CERTIFICAUTH
1094	AUTOREGISTER
1095	AUTOAUTH
1096	HTTP
1097	IIOP
1098	ECI
1099	VERIFY
1100	BOTH
1101	CORBA
1102	EJB
1103	HOME
1104	ASSERTED
1105	KERBEROS
1106	SUPPORTED
1107	OTS
1109	RESPECTED
1110	NOPHASEOUT
1111	PHASEOUT
1112	INITING
1113	PENDINIT

Wert	CVDA
1114	PENDRESOLVE
1115	RESOLVING
1116	UNRESOLVED
1117	UNUSABLE
1118	DISCARDING
1119	ATOM
1120	AUTOTERMID
1121	URMTERMID
1122	CLIENT
1123	SERVER
1124	PIPELINE
1125	DISABLEDHOST
1126	ANALYZER
1127	NOANALYZER
1128	EXTENDED
1129	NOTEXTENDED
1130	COMPRESS
1131	NOCOMPRESS
1140	AUTOINSTALL
1141	CORBASERVER
1142	DB2CONN
1143	DB2ENTRY
1144	DISPATCHER
1145	DOCTEMPLATE
1146	ENQUEUE
1147	JOURNALNAME
1148	JOURNALNUM
1149	JVMPPOOL
1150	JVMPROFILE
1151	JVMPROGRAM
1152	LSRPOOL
1153	MONITOR
1154	MVSTCB
1155	NODE
1156	RECOVERY

Wert	CVDA
1157	REQUESTMODEL
1158	STATS
1159	STORAGE
1160	STREAMNAME
1161	SUBPOOL
1162	SYSDUMPCODE
1163	TABLEMGR
1164	TASKSUBPOOL
1166	TCIPSERVICE
1168	TEMPLATENAME
1169	TRANCLASS
1170	TRANDUMPCODE
1171	TRANSACTION
1173	URIMAP
1174	WEBSERVICE
1175	MQCONN
1176	IPCONN
1177	LIBRARY
1178	PROGRAMDEF
1179	ATOMSERVICE
1180	BUNDLE
1182	PARTNER
1183	TSMODEL
1184	DBTRAN
1185	JOURNALMODEL
1186	TYPETERM
1187	SESSIONS
1188	ENQMODEL
1189	DJAR
1190	PROCESSTYPE
1191	EVENTBINDING
1192	EVENTPROCESS
1193	JVMSERVER
1194	XMLTRANSFORM
1196	EPADAPTER

Wert	CVDA
1197	MESSAGEID
1198	LOGREPLICATE
1199	NOLOGREPLICA
1200	EPADAPTERSET
1201	RESIDENT
1202	NONRESIDENT
1203	UTC
1204	SOMEAVAIL
1206	DSIE
1207	MQMONITOR
1208	NOAUTOSTART
1209	ASRUNTRAN
1210	RREPL
1211	REPLICATOR
1212	NOREPLICATOR
1214	SECERROR
1215	NODEJSAPP
1216	SECURITY
1217	JWT
1218	DUMPCODE
1219	TABLEONLY
1220	PATCH
1221	OSGI
1222	LIBERTY
1223	DUMP
1224	GATHER
1225	REFRESHPKGS
1226	SERVERDUMP
1227	REFRESH
1228	JAVACORE
1229	HEAP
1230	SNAPTRACE
1231	DIAGNOSTICS
1232	CONFIG

Anmerkung: VTAM ist jetzt z/OS Communications Server.

Kapitel 5. Landessprachencodes für die Anwendungsentwicklung

Sprachencodes werden in Form eines Zeichens für die Optionen NATLANG und NATLANGINUSE sowie in Form von drei Zeichen für die Optionen LANGUAGECODE und LANGINUSE in API-Befehlen angegeben.

Tabelle 20. CICS-Sprachsuffixe		
Suffix	IBM Code	Sprachenname
A	ENG	Englisch (GB)
B	PTB	Portugiesisch (Brasilien)
C	CHS	Vereinfachtes Chinesisch
D	DAN	Dänisch
E	ENU	US-Englisch
F	FRA	Französisch
G	DEU	Deutsch
H	KOR	Koreanisch
I	ITA	Italienisch
J	ISL	Isländisch
K	JPN	Japanisch
L	BGR	Bulgarisch
M	MKD	Mazedonisch
N	NOR	Norwegisch
O	ELL	Griechisch
P	PTG	Portugiesisch
Q	ARA	Arabisch
R	RUS	Russisch
S	ESP	Spanisch
T	CHT	Traditionelles Chinesisch
U	UKR	Ukrainisch
V	SVE	Schwedisch
W	FIN	Finnisch
X	HEB	Hebräisch
Y	SHC	Serbokroatisch (Kyrillisch)
Z	THA	Thailändisch
1	BEL	Weißrussisch
2	CSY	Tschechisch
3	HRV	Kroatisch

Tabelle 20. CICS-Sprachsuffixe (Forts.)

Suffix	IBM Code	Sprachenname
4	HUN	Ungarisch
5	PLK	Polnisch
6	ROM	Rumänisch
7	SHL	Serbokroatisch (lateinischer Zeichensatz)
8	TRK	Türkisch
9	NLD	Niederländisch

Es gibt weitere IBM Codes, die von CICS nicht unterstützt werden.

Tabelle 21. Weitere IBM Sprachencodes

IBM Code	Sprachenname
AFR	Afrikaans
CAT	Katalanisch
DES	Deutsch (Schweiz)
ENA	Englisch (Australien)
ENP	Englisch (Großschreibung)
FRB	Französisch (Belgien)
FRC	Französisch (Kanada)
FRS	Französisch (Schweiz)
GAE	Gälisch (Irland)
ITS	Italienisch (Schweiz)
NLB	Niederländisch (Belgien) - Flämisch
NON	Nynorsk-Norwegisch
RMS	Rätoromanisch
SKY	Slowakisch
SLO	Slowenisch
SRL	Serbisch (lateinischer Zeichensatz)
SRB	Serbisch (Kyrillisch)
SQI	Albanisch
URD	Urdu

Kapitel 6. Terminalsteuerung

Diese Referenzinformationen gelten für alle Terminals und logischen Einheiten. Genauere Informationen hierzu finden Sie in den Befehlsbeschreibungen.

Befehle und Optionen für Terminals und logische Einheiten

In diesem Abschnitt sind die Befehle und Optionen beschrieben, die für Terminals und logische Einheiten gelten.

Vollwortlängen

Bei allen Terminalsteuerungsbefehlen können Vollwortlängenoptionen anstelle von Halbwortlängenoptionen verwendet werden. Insbesondere bei Verwendung der folgenden Optionen in Befehlen CONVERSE, RECEIVE oder SEND können stattdessen die entsprechenden Alternativen angegeben werden (Ausnahmen sind entsprechend gekennzeichnet):

Option	Alternative
LENGTH	FLENGTH
TOLENGTH	TOFLENGTH
FROMLENGTH	FROMFLENGTH
MAXLENGTH	MAXFLENGTH

Anwendungsprogramme müssen bei der Verwendung von Vollwort- und Halbwortoptionen für Terminalsteuerungsbefehle konsistent sein. Der Maximalwert, der als Parameter für ein beliebiges Längenschlüsselwort angegeben werden kann, ist 32767.

Daten aus Terminal oder logischer Einheit lesen (RECEIVE)

Mit dem Befehl RECEIVE werden Daten aus einem Terminal oder einer logischen Einheit gelesen. Die Option INTO wird zur Angabe des Bereiches verwendet, in den die Daten eingefügt werden sollen. Alternativ kann in der Option SET ein Zeigerverweis angegeben werden. CICS fordert einen Bereich an, der groß genug für die Daten ist, und legt den Zeigerverweis auf die Adresse dieser Daten fest.

Der Inhalt dieses Bereichs ist für die Task bis zum nächsten Ein-/Ausgabebefehl für das Terminal verfügbar. Der Bereich gehört jedoch nicht zur Task und wird von CICS bei der Verarbeitung der nächsten Anforderung freigegeben. Daher kann dieser Bereich nicht zur weiteren Verarbeitung an CICS zurück übergeben werden.

Mit der Option MAXLENGTH kann die Anwendung die maximale Länge der Daten angeben, die das Programm akzeptiert. Falls die Option MAXLENGTH in einem Befehl RECEIVE weggelassen wird, für den die Option INTO angegeben ist, kann die maximale Länge der vom Programm akzeptierten Daten in der Option LENGTH angegeben werden. Wird die Option MAXLENGTH in einem Befehl RECEIVE weggelassen, für den die Option SET angegeben ist, fordert CICS genügend Speicher für alle verfügbaren Daten an.

Falls die Daten die angegebene maximale Länge überschreiten und die Option NOTRUNCATE angegeben ist, werden die verbleibenden Daten für die Erfüllung nachfolgender Befehle RECEIVE verfügbar gemacht. Ist die Option NOTRUNCATE nicht angegeben, werden die Daten abgeschnitten und die Bedingung LENGERR tritt auf. In diesem Fall wird bei angegebener Option LENGTH der benannte Datenbereich auf die tatsächliche Datenlänge (vor dem Abschneiden) gesetzt, wenn Daten empfangen wurden. Der erste Befehl RECEIVE in einer durch ein Terminal gestarteten Task gibt keinen Lesevorgang für die Terminalsteuerung aus, sondern kopiert den Eingabepuffer, auch wenn die Datenlänge Null ist. Um den Lesevorgang der Terminalsteuerung auszulösen, muss ein zweiter Befehl RECEIVE ausgegeben werden.

Wenn durch den Systeminitialisierungsparameter PRINT eine Programmabrufaste als Drucktaste definiert ist und diese Taste als Reaktion auf einen Befehl RECEIVE gedrückt wird, hat dies keinen Einfluss auf das Anwendungsprogramm. Der Befehl RECEIVE wird erfüllt und die Fortsetzung der Anwendung ist

zulässig, sobald ein weiterer Abruf (also mit einer der anderen Programmabrufstasten, einer der Funktionstasten, der Eingabetaste oder dem Lichtstift) über die Tastatur erfolgt.

Daten in Terminal oder logische Einheit schreiben (SEND)

Mit dem Befehl SEND werden Daten in ein Terminal oder eine logische Einheit geschrieben. Die Optionen FROM und LENGTH geben den Datenbereich, aus dem die Daten entnommen werden sollen, und die Länge der Daten (in Byte) an. Bei einer durch die automatische Transaktionsinitialisierung (ATI) gestarteten Transaktion sollte dem ersten Befehl RECEIVE in einer Transaktion immer ein Befehl SEND vorausgehen.

Sofern nicht ebenfalls die Option WAIT des Befehls SEND angegeben ist, wird die Übertragung der Daten, die dem Befehl SEND zugeordnet sind, verzögert, bis ein späteres Ereignis (z. B. ein Synchronisationspunkt) stattfindet. Diese verzögerte Übertragung verringert den Datenfluss, weil sie die Übertragung von Datenflusssteuerungen zusammen mit den Daten ermöglicht.

Die Übertragung wird für die verteilte Transaktionsverarbeitung nicht verzögert, wenn die regionsübergreifende Kommunikation verwendet wird.

Ein-/Ausgabe eines Terminals für eine Transaktion synchronisieren (WAIT TERMINAL)

Mit diesem Befehl wird sichergestellt, dass eine Terminaloperation abgeschlossen worden ist, bevor in einer Task, unter der mehrere Operationen für Terminals oder logische Einheiten ausgeführt werden, eine weitere Verarbeitung stattfindet. Alternativ kann die Option WAIT in einem Befehl SEND angegeben werden. (Für einen Befehl RECEIVE wird immer ein Wartestatus ausgeführt.) Beide Methoden können bewirken, dass die Ausführung einer Task ausgesetzt wird. Falls die Aussetzung erforderlich ist, wird die Steuerung an CICS zurückgegeben. Die Ausführung der Task wird wiederaufgenommen, sobald die Operation abgeschlossen worden ist.

Selbst wenn die Option WAIT in einem Befehl SEND nicht angegeben ist, stellt das EXEC-Schnittstellenprogramm sicher, dass die Operation abgeschlossen worden ist, bevor ein nachfolgender Befehl RECEIVE oder SEND ausgegeben wird.

Dialog mit Terminal oder logischer Einheit (CONVERSE)

Für die meisten Terminals oder Typen von logischen Einheiten kann für die Kommunikation ein Dialogbetrieb verwendet werden. Zu diesem Zweck wird der Befehl CONVERSE verwendet; er bedeutet, dass das 3650-Anwendungsprogramm mit dem Hostprozessor kommuniziert. Falls diese Option nicht angegeben ist, kann das 3650-Anwendungsprogramm nicht mit dem Hostprozessor kommunizieren. Im Allgemeinen ist ein Befehl CONVERSE mit einer Kombination vergleichbar, bei der auf einen Befehl SEND unmittelbar ein Befehl WAIT TERMINAL und dann ein Befehl RECEIVE folgt. Für den Befehl CONVERSE sind jedoch nicht alle Optionen der Befehls SEND und RECEIVE gültig; spezifische Regeln sind in den Syntaxbeschreibungen für die unterschiedlichen Einheiten angegeben. Die Option TOLENGTH ist äquivalent zur Option LENGTH des Befehls RECEIVE und die Option FROMLENGTH ist äquivalent zur Option LENGTH des Befehls SEND.

Asynchronen Interrupt senden (ISSUE SIGNAL)

Mit diesem Befehl wird in einer Transaktion im Empfangsmodus der sendenden Transaktion signalisiert, dass eine Modusänderung erforderlich ist. Die Ausführung des Befehls löst die Bedingung SIGNAL beim nächsten Befehl SEND oder RECEIVE aus, der in der sendenden Transaktion ausgeführt wird; mit einem zuvor für diese Bedingung ausgeführten Befehl HANDLE CONDITION kann dann die Anforderung entweder ausgeführt oder ignoriert werden.

Wählleitung trennen (ISSUE DISCONNECT)

Mit diesem Befehl wird eine Leitungsverbindung zwischen einem Terminal und dem Prozessor oder eine Sitzung zwischen logischen SNA-Einheiten (LUs) getrennt, sobald die Transaktion abgeschlossen ist. Falls es sich bei dem Terminal um eine Einheit mit Puffern handelt, gehen die Daten in den Puffern verloren.

Bei Verwendung für eine logische SNA-Einheit bewirkt der Befehl ISSUE DISCONNECT, der erst nach Abschluss der Task wirksam wird, die Abmeldung des Terminals, die Freigabe des Kommunikationsbereichs,

die Löschung des nächsten TRANID-Wertes, den Stopp des BMS-Seitenwechsels und - bei wirksamer automatischer Installation - die Löschung der Terminaldefinition.

Programmierung für Fernschreiber

Der Fernschreiber (nur Welthandel) verwendet zwei unterschiedliche Steuerzeichen für die Druckformatierung.

Diese Steuerzeichen sind folgende:

<	Wagenrücklauf (X'22' in ITA2-Code oder X'15' in EBCDIC)
≡	Zeilenvorschub (X'28' in ITA2-Code oder X'25' in EBCDIC)

Verwenden Sie das Zeichen < zuerst; andernfalls werden nachfolgende Zeichen (Daten) möglicherweise gedruckt, während die Typenstange nach links bewegt wird.

Nachrichtenformat

Damit eine neue Nachricht in einer neuen Zeile am linken Rand anfängt, beginnen Sie den Nachrichtentext mit dem Zeichen X'1517' (EBCDIC). CICS erkennt das Zeichen X'17' und ändert es in X'25' (X'17' ist ein nicht darstellbares Zeichen).

Damit im Nachrichtenhauptteil bei einer einzigen Übertragung mehrere Zeilen geschrieben werden, müssen die Zeilen durch das Zeichen X'1525' bzw. - falls mehrere Leerzeilen erforderlich sind - durch das Zeichen X'152525...25' voneinander getrennt werden.

Um die Eingabe der nächsten Nachricht in einer neuen Zeile am linken Rand zu ermöglichen, muss die vorausgehende Nachricht mit dem Zeichen X'1517' enden. CICS erkennt das Zeichen X'15' und ändert das darauf folgende Zeichen in X'25'.

Falls zwei oder mehr aufeinanderfolgende Ausgabenachrichten einen Nachrichtenanfang (message begin) und ein Nachrichtenende (message end) mit demselben Erscheinungsbild aufweisen, darf das nächste Zeichen nach dem letzten Zeichen des Nachrichtenendes nicht X'15' sein, damit das Nachrichtenende der vorausgehenden Nachricht vom Nachrichtenanfang der nächsten Nachricht unterschieden werden kann.

Nachrichtenlänge

Nachrichten für Fernschreiberterminals sollten eine Länge von ca. 3000 Byte oder ungefähr 300 Wörtern nicht überschreiten.

Verbindung über z/OS Communications Server

Sowohl eine Einheit 'TWX Modell 33/35 Common Carrier Teletypewriter Exchange' als auch eine Einheit 'WTTY Teletypewriter' (nur Welthandel) können mit CICS über z/OS Communications Server unter Verwendung von NTO verbunden werden.

Falls eine Einheit über z/OS Communications Server unter Verwendung von NTO verbunden ist, werden dieselben Protokolle wie bei der logischen 3767-Einheit verwendet und das Anwendungsprogramm kann diese Protokolle nutzen (beispielsweise für HANDLE CONDITION SIGNAL). Der Datenstrom wird jedoch nicht in einen 3767-Datenstrom umgesetzt, sondern bleibt ein Datenstrom für eine TWX-/WTTY-Einheit.

Operationen für Bildschirmeinheiten

Neben den Standardterminalsteuerungsbefehlen für das Senden und Empfangen von Daten werden zur Verwendung bei Bildschirmeinheiten wie beispielsweise der 3270-Einheit mehrere Befehle und Listen bereitgestellt.

Dies sind die folgenden Befehle:

- Angezeigte Informationen drucken (ISSUE PRINT)
- Angezeigte Informationen kopieren (ISSUE COPY)
- Alle ungeschützten Felder entfernen (ISSUE ERASEAUP)
- Eingabe ohne Daten verarbeiten (RECEIVE)
- AID-Zeichen verarbeiten (HANDLE AID)

Außerdem werden die folgenden Listen zur Verfügung gestellt:

- Liste der Standard-AID-Zeichen (DFHAID)
- Liste der Standardattribut- und Druckersteuerzeichen (DFHBMSCA)

Für Einheiten mit umschaltbarer Anzeigegröße werden die zu verwendende Anzeigegröße und die für eine bestimmte Transaktion zu verwendende Größe durch die CICS-Tabellengenerierung definiert. Diese Werte können mit dem Befehl ASSIGN abgerufen werden (siehe „ASSIGN“ auf Seite 42).

Beziehen Sie in den ersten Befehl SEND immer die Option ERASE ein, damit der Inhalt der Anzeige gelöscht und die Anzeige entsprechend den übertragenen Daten formatiert wird. Dieser erste Befehl SEND mit der Option ERASE wählt außerdem die zu verwendende Anzeigegröße aus, die mithilfe der RDO-Option SCRNSIZE oder in der Terminalsteuertabelle angegeben wird. Falls die Option ERASE nicht angegeben ist, ist die Anzeigegröße mit der vorherigen Einstellung identisch, was falsch sein kann.

Verwenden Sie außerhalb einer Transaktion die Löschtaste, um die Anzeige auf ihre Standardgröße zu setzen.

Angezeigte Informationen drucken (ISSUE PRINT)

Der Befehl ISSUE PRINT druckt angezeigte Daten auf dem ersten verfügbaren Drucker, der für die Beantwortung einer Druckanforderung infrage kommt.

Bei einer logischen 3270-Einheit oder einer logischen Einheit mit 3650- bzw. 3270-Hostdialog handelt es sich hierbei um einen Drucker, der durch die RDO-Optionen PRINTER und ALTPRINTER für TERMINAL definiert ist, oder um einen Drucker, der durch das Benutzerprogramm für die automatische Installation bereitgestellt wird.

Bei einer logischen Einheit mit 3270-Anzeige und Druckeradapterfunktion (RDO-Option PRINTADAPTER(YES) für TYPETERM), die mit einer 3274- oder 3276-Einheit verwendet wird, handelt es sich hierbei um einen Drucker, der durch die Druckerberechtigungsmatrix zugeordnet wird. Weitere Informationen enthält das Dokument *An Introduction to the IBM 3270 Information Display System*.

Bei einer logischen Einheit mit 3790- bzw. 3270-Anzeige handelt es sich um einen Drucker, der durch die 3790-Einheit zugeordnet wird.

Damit ein Drucker verfügbar ist, muss er in Betrieb sein und darf gegenwärtig nicht an eine Task angehängt sein.

Damit eine logische 3270-Einheit zulässig ist, muss sie durch die RDO-Optionen PRINTER und ALTPRINTER für TERMINAL oder durch einen Drucker angegeben worden sein, der durch das Benutzerprogramm für die automatische Installation bereitgestellt wurde, und die korrekte Pufferkapazität besitzen. Falls in der RDO-Definition für TYPETERM ebenfalls die Kopierfunktion angegeben ist (COPY(YES)), muss sich der Drucker in demselben Controller befinden.

Falls ein Befehl ISSUE PRINT ausgeführt wird, muss das CICS-System, das Eigner des Terminals ist, von dem die Transaktion ausgeführt wird, ebenfalls Eigner des beteiligten Druckers sein.

Bei einigen 3270-Anzeigen ist es möglich, die angezeigten Informationen auch ohne Verwendung von CICS zu drucken.

Angezeigte Informationen kopieren (ISSUE COPY)

Mit dem Befehl ISSUE COPY werden die im Puffer eines angegebenen Terminals enthaltenen Daten und das zugehörige Format in den Puffer des Terminals kopiert, von dem die Transaktion gestartet wurde.

Für eine Verbindung mit LU-Typ 2 kann dieser Befehl nicht verwendet werden. Beide Terminals müssen an denselben fernen Controller angehängt sein. Das Terminal, dessen Puffer kopiert werden soll, ist in der

Option TERMID angegeben. Falls die Terminal-ID nicht gültig ist (also nicht in der Terminalsteuertabelle vorhanden ist), tritt die Bedingung TERMIDERR auf. Die auszuführende Kopierfunktion wird durch das Kopiensteuerungszeichen (Copy Control Character, CCC) definiert, das in der Option CTLCHAR des Befehls ISSUE COPY angegeben ist.

Die Option WAIT des Befehls ISSUE COPY stellt sicher, dass die Operation abgeschlossen worden ist, bevor die Steuerung an das Anwendungsprogramm zurückgegeben wird.

Alle ungeschützten Felder entfernen (ISSUE ERASEAUP)

Mit dem Befehl ISSUE ERASEAUP werden alle ungeschützten Felder eines 3270-Puffers entfernt.

Zu diesem Zweck werden die folgenden Aktionen eingesetzt:

1. Alle ungeschützten Felder werden auf Nullen (X'00') gesetzt.
2. Die Tags für modifizierte Daten (MDTs) in jedem ungeschützten Feld werden auf Null zurückgesetzt.
3. Der Cursor wird am ersten ungeschützten Feld positioniert.
4. Die Tastatur wird wiederhergestellt.

Die Option WAIT des Befehls ISSUE ERASEAUP stellt sicher, dass die Operation abgeschlossen worden ist, bevor die Steuerung an das Anwendungsprogramm zurückgegeben wird.

Eingabe ohne Daten verarbeiten (RECEIVE)

Die Verwendung des Befehls RECEIVE ohne Optionen bewirkt, dass eine Eingabe stattfindet und der EXEC-Schnittstellenblock (EIB) aktualisiert wird.

Durch CICS empfangene Daten werden nicht an das Anwendungsprogramm übergeben und gehen verloren. Ein Wartestatus wird impliziert. Im EXEC-Schnittstellenblock werden unter anderem die Felder EIBC-POSN und EIBAID aktualisiert.

Cursorposition (EIBCPOSN)

Für jede Eingabeoperation der Terminalsteuerung (oder BMS), die einer Bildschirmeinheit zugeordnet ist, wird die Adresse (Position) des Anzeigecursors im Feld EIBCPOSN des EXEC-Schnittstellenblocks angeordnet. Die Cursoradresse hat das Format eines Halbwort-Binärwertes und bleibt erhalten, bis sie durch eine neue Eingabeoperation aktualisiert wird.

AID-Zeichen

Für jede Eingabeoperation der Terminalsteuerung (oder BMS), die einer Bildschirmeinheit zugeordnet ist, wird das AID-Zeichen im Feld EIBAID des EXEC-Schnittstellenblocks angeordnet.

Das AID-Zeichen gibt an, mit welcher Methode der Terminalbediener die Übertragung von Informationen aus der Einheit an CICS ausgelöst hat, beispielsweise mit der Eingabetaste, einer Funktionstaste, dem Lichtstift usw. Das Feld bleibt unverändert, bis es durch eine neue Eingabeoperation aktualisiert wird.

Das Feld EIBAID kann nach jeder Eingabeoperation für die Terminalsteuerung (oder BMS) getestet werden, um die weitere Verarbeitung zu bestimmen; zu diesem Zweck wird eine Standard-AID-Liste (DFHAID) bereitgestellt. Alternativ kann mit dem Befehl HANDLE AID die Steuerung an angegebene Kennsätze übergeben werden, wenn die AID-Zeichen empfangen werden.

Die Felder EIBAID und EIBCPOSN werden ebenfalls bei der Initialisierung von Nicht-ATI-Tasks sowie nach jeder Eingabe für die Terminalsteuerung und BMS aktualisiert.

Kapitel 7. Common Programming Interface Communications (CPI Communications)

Common Programming Interface Communications (CPI Communications) ist das Kommunikationselement von Systems Applications Architecture (SAA) Common Programming Interface (CPI).

CPI Communications in CICS stellt eine alternative Anwendungsprogrammierschnittstelle (API) zur vorhandenen CICS-Unterstützung für Advanced Program-to-Program Communications (APPC) bereit. CPI Communications stellt eine verteilte Transaktionsverarbeitung (DTP - Distributed Transaction Processing) für APPC-Sitzungen bereit und kann in Assemblersprache, COBOL, PL/I oder C verwendet werden.

CPI Communications definiert eine API, die in APPC-Netzen verwendet werden kann, die mehrere Systemplattformen einschließen, in denen die Konsistenz einer gemeinsamen API von Vorteil ist.

Die CPI Communications-Schnittstelle kann mit Anwendungen auf jedem System kommunizieren, das eine APPC-API bereitstellt. Dazu gehören Anwendungen auf CICS-Plattformen. Sie können APPC-API-Befehle an einem Ende eines Dialogs und CPI Communications-Befehle am anderen Ende verwenden. CPI Communications erfordert bestimmte Informationen (Nebeninformationen) zum Starten eines Dialogs mit einem Partnerprogramm. Die CICS-Implementierung von Nebeninformationen wird durch die Verwendung der Partnerressource erreicht, die von Ihrem Systemprogrammierer verwaltet werden muss.

Aufrufe aus der Anwendung an die CPI Communications-Schnittstelle werden durch die Bearbeitung von Programmverbindungen mit dem CICS CPI Communications-Stub (DFHCPLC) aufgelöst. Informationen dazu finden Sie unter [Von CICS bereitgestellte Schnittstellenmodule einschließen](#).

Die CPI Communications-API ist als allgemeine Aufrufchnittstelle definiert. Die Schnittstelle wird in [z/VM: CPI Communications User's Guide](#) beschrieben.

SAA Resource Recovery

SAA Resource Recovery ist das Wiederherstellungselement der Systems Application Architecture (SAA) Common Programming Interface (CPI).

SAA Resource Recovery stellt die alternative Anwendungsprogrammierschnittstelle (API) dieser Architektur für die Funktionen EXEC CICS SYNCPOINT und EXEC CICS SYNCPOINT ROLLBACK in CICS bereit. Weitere Informationen finden Sie unter [Systems Application Architecture Common Programming Interface Resource Recovery Reference](#).

CICS unterstützt nur diejenigen Rückgabecodes von SAA Resource Recovery, die mit vorhandenen EXEC CICS-Befehlen übereinstimmen. Daher verbleiben nur die beiden Rückgabecodes RR_OK und RR_BACKED_OUT.

SRRCMT

Für den Commitaufruf SRRCMT von SAA Resource Recovery (das Äquivalent zu EXEC CICS SYNCPOINT) gibt es die folgenden Rückgabecodes:

- RR_OK
- RR_COMMITTED_OUTCOME_PENDING
- RR_COMMITTED_OUTCOME_MIXED
- RR_PROGRAM_STATE_CHECK
- RR_BACKED_OUT
- RR_BACKED_OUT_OUTCOME_PENDING
- RR_BACKED_OUT_OUTCOME_MIXED

Aufgrund der beschriebenen Einschränkung werden diese Rückgabecodes durch die folgenden Codes ersetzt:

- RR_COMMITTED_OUTCOME_PENDING, RR_OK
- RR_COMMITTED_OUTCOME_MIXED, RR_OK
- RR_PROGRAM_STATE_CHECK (wird als Abbruchcode ASP2 angezeigt)
- RR_BACKED_OUT_OUTCOME_PENDING, RR_BACKED_OUT
- RR_BACKED_OUT_OUTCOME MIXED, RR_BACKED_OUT

SRRBACK

Für den Backoutaufruf SRRBACK von SAA Resource Recovery (das Äquivalent zu EXEC CICS SYNCPOINT ROLLBACK) gibt es die folgenden Rückgabecodes:

- RR_OK
- RR_COMMITTED_OUTCOME_PENDING
- RR_COMMITTED_OUTCOME_MIXED

Aufgrund der beschriebenen Einschränkung werden alle diese Rückgabecodes durch den Code RR_OK ersetzt.

Kapitel 8. BMS-spezifische Konstanten

In diesem Abschnitt werden die BMS-spezifischen Standardattribute und Druckersteuerzeichen, eine Bit-map für Attribute, MSR-Steuerwertkonstanten und AID-Zeichenkonstanten beschrieben.

Die Standardliste DFHBMSCA vereinfacht die Angabe von Feldattributen und Druckersteuerzeichen. In [Tabelle 22 auf Seite 841](#) sind die symbolischen Namen für die verschiedenen Kombinationen von Attributen und Steuerzeichen aufgelistet. Wenn Sie andere als die gezeigten Kombinationen benötigen, müssen Sie sie separat generieren. Zu diesem Zweck finden Sie in [Tabelle 23 auf Seite 843](#) eine Bitmap von Attributen. Den Wert einer Attributkonstanten finden Sie in der Veröffentlichung *3274 Control Unit Reference Summary GA27-2827*.

Sie können die Standardliste für Attribute und Druckersteuerzeichen abrufen, indem Sie das Copybook DFHBMSCA in Ihre Anwendung kopieren.

- Für COBOL-Benutzer besteht die Liste aus einer Gruppe von 01-Anweisungen, die in den Arbeitsspeicherabschnitt (WORKING STORAGE SECTION) kopiert werden können.
- Für C-Benutzer wird sie wie folgt in Anwendungen eingeschlossen:

```
#include "dfhbmsca.h"
```

- Für PL/I-Benutzer besteht sie aus DECLARE-Anweisungen, die elementare Zeichenvariablen definieren.
- Für Benutzer der Assemblersprache besteht die Liste aus einer Gruppe von EQU-Anweisungen.

Sie müssen den symbolischen Namen DFHDFT in der Anwendungsstruktur verwenden, um ein Mapattribut mit dem Standardwert zu überschreiben. Durch einen hohen Wert wie 'X'FF' können Sie die Attribute COLOR, HILIGHT, OUTLINE, PS, SOSI oder VALIDN auf ihre Standardwerte zurücksetzen. Zur Angabe von Standardwerten in einer Sequenz von Attributfestlegungen (Set-Attribute-Sequence) im Textbuild verwenden Sie die symbolischen Namen DFHDFCOL, DFHBASE oder DFHDFHI.

Tabelle 22. Standardliste für Attribute und Druckersteuerzeichen: DFHBMSCA

Konstante	Bedeutung
DFHBMPEM	Drucker: Nachrichtenende
DFHBMPNL	Drucker: Zeilenvorschubzeichen
DFHBMPFF	Drucker: Seitenvorschub
DFHBMPCR	Drucker: Rücklauf
DFHBMASK	Automatisches Überspringen (Autoskip)
DFHBMUNP	Ungeschützt
DFHBMUNN	Ungeschützt und numerisch
DFHBMPRO	Geschützt
DFHMBRY	Hell
DFHBMDAR	Dunkel
DFHBMFSE	MDT (Modified Data Tag - Tag für modifizierte Daten) gesetzt
DFHBMPRF	Geschützt und MDT gesetzt
DFHBMAF	Automatisches Überspringen und MDT gesetzt
DFHBMAFB	Automatisches Überspringen und Hell
DFHBMPSO	DBCS-Startzeichen X'OE'

Tabelle 22. Standardliste für Attribute und Druckersteuerzeichen: DFHBMSCA (Forts.)

Konstante	Bedeutung
DFHBMPSI	DBCS-Endezeichen X'0F'
DFHBMEOF	Feld gelöscht
DFHBMCUR	Feld mit Cursor markiert
DFHBMEC	Gelöschtes Feld mit Cursor (nur COBOL)
DFHBMFLG	Flags (nur COBOL)
DFHBMDET	Feld erkannt
DFHSA ¹	Attributfestlegungsanweisung (Set-Attribute-Anweisung)
DFHERROR	Fehlercode
DFHCOLOR ¹	Farbe
DFHPS ¹	Programmierte Symbole
DFHHLT ¹	Hervorheben
DFH3270 ¹	3270-Basisfeldattribut
DFHVAL	Validierung
DFHOUTLN	Code für Feldumrandungsattribut
DFHBKTRN	Code für Hintergrundtransparenzattribut
DFHALL ¹	Alles auf Standardwerte zurücksetzen
DFHDFT	Standardwert
DFHDFCOL ¹	Standardfarbe
DFHBLUE	Blau
DFHRED	Rot
DFHPINK	Rosa
DFHGREEN	Grün
DFHTURQ	Türkis
DFHYELLO	Gelb
DFHNEUTR	Neutral
DFHBASE ¹	Programmierte Basissymbole
DFHDFHI ¹	Normal
DFHBLINK	Blinken
DFHREVR	Umkehranzeige
DFHUNDLN	Unterstreichung
DFHMFIL ²	Obligatorisches Füllen
DFHMENT ²	Obligatorische Eingabe
DFHMFEE	Obligatorisches Füllen und obligatorische Eingabe
DFHMT	Trigger (Auslöser)
DFHMFT	Obligatorisches Füllen und Trigger
DFHMET	Obligatorische Eingabe und Trigger

Tabelle 22. Standardliste für Attribute und Druckersteuerzeichen: DFHBMSCA (Forts.)

Konstante	Bedeutung
DFHMFET	Obligatorisches Füllen, obligatorische Eingabe und Trigger
DFHUNNOD	Ungeschützt, Anzeigeunterdrückung, Druckunterdrückung, nicht erkennbar, MDT
DFHUNIMD	Ungeschützt, Intensivanzeige, lichtstiftempfindlich, MDT
DFHUNNUM	Ungeschützt, numerisch, MDT
DFHUNNUB	Ungeschützt, numerisch, Intensivanzeige, lichtstiftempfindlich
DFHUNINT	Ungeschützt, numerisch, Intensivanzeige, lichtstiftempfindlich, MDT
DFHUNNON	Ungeschützt, numerisch, Anzeigeunterdrückung, Druckunterdrückung, nicht erkennbar, MDT
DFHPROTI	Geschützt, Intensivanzeige, lichtstiftempfindlich
DFHPROTN	Geschützt, Anzeigeunterdrückung, Druckunterdrückung, nicht erkennbar
DFHDFFR	Standardumrandung
DFHUNDER	Unterstreichung
DFHRIGHT	Rechte vertikale Linie
DFHOVER	Überstreichung
DFHLEFT	Linke vertikale Linie
DFHBOX	Unterstreichung, rechte vertikale Linie, Überstreichung und linke vertikale Linie
DFHSOSI	SOSI=yes
DFHTRANS	Hintergrundtransparenz
DFHOPAQ	Keine Hintergrundtransparenz

Hinweise:

¹ Nur für Textverarbeitung. Verwenden Sie diese Konstanten zum Erstellen von eingebetteten Attributfestlegungsanweisungen in Benutzertext.

² Kann in Attributfestlegungsanweisungen nicht verwendet werden.

Tabelle 23. Bitmap für Attribute

prot	a/n	hi	spd	ndp	mdt	ebcd	ascii	char
U						40	20	b (blank - leer)
U					Y	C1	41	A
U			Y			C4	44	D
U			Y		Y	C5	45	E
U		H	Y			C8	48	H
U		H	Y		Y	C9	49	I
U				Y		4C	3C	<
U				Y	Y	4D	28	(
U	N					50	26	
U	N				Y	D1	4A	J

Tabelle 23. Bitmap für Attribute (Forts.)

prot	a/n	hi	spd	ndp	mdt	ebcd	ascii	char
U	N		Y			D4	4D	M
U	N		Y		Y	D5	4E	N
U	N	H	Y			D8	51	Q
U	N	H	Y		Y	D9	52	R
U	N			Y		5C	2A	*
U	N			Y	Y	5D	29)
P						60	2D	- (Bindestrich)
P					Y	61	2F	/
P			Y			E4	55	U
P			Y		Y	E5	56	V
P		H	Y			E8	59	Y
P		H	Y		Y	E9	5A	Z
P				Y		6C	25	%
P				Y	Y	6D	5F	_ (Unterstrich)
P	S					F0	30	0
P	S				Y	F1	31	1
P	S		Y			F4	34	4
P	S		Y		Y	F5	35	5
P	S	H	Y			F8	38	8
P	S	H	Y		Y	F9	39	9
P	S			Y		7C	40	@
P	S			Y	Y	7D	27	'

Tabelle 24. Schlüssel für Attribute und Einstellungen in einer Bitmap

Code	Bedeutung
a/n	Automatisches Überspringen oder numerisch
ascii	ASCII (American National Standard Code for Information Interchange)
char	Zeichen, das dem Hexadezimalcode entspricht
ebcd	EBCDIC (Extended Binary Coded Decimal Interchange Code)
hi	Intensivanzeige
H	Hoch
mdt	Tag für modifizierte Daten
ndp	Anzeigeunterdrückung in Ausgabe
N	Numerisch
prot	Geschützt
P	Geschützt

Tabelle 24. Schlüssel für Attribute und Einstellungen in einer Bitmap (Forts.)	
Code	Bedeutung
spd	Selector pen detectable (lichtstiftempfindlich)
S	Automatic Skip (automatisches Überspringen)
U	Ungeschützt
Y	Yes (Ja)

Steuerwertkonstanten für Magnetcodeleser (DFHMSRCA)

Eine Auswahl von Steuerwertkonstanten für Magnetcodeleser (MSR - Magnetic Slot Readers) wurde für CICS erstellt und im Copybook DFHMSRCA gespeichert. Die Muster werden als benannte Konstanten gespeichert, die durch einfache Anwendungsprogrammbefehle geladen werden können. Die Bereitstellung solcher Konstanten erspart dem Programmierer das Erstellen eines allgemein verwendeten Bitmusters, wann immer es erforderlich ist.

Werte für MSR-Steuerbyte

Eine Auswahl von Steuerbytewerten für Magnetcodeleser (MSR - Magnetic Slot Reader) wurde für CICS erstellt und im Copybook DFHMSRCA gespeichert. In der folgenden Tabelle wird die Bedeutung der einzelnen Bit erläutert.

Die in DFHMSRCA bereitgestellten Konstanten sind in [Tabelle 25 auf Seite 845](#) aufgelistet.

Tabelle 25. Standardliste DFHMSRCA	
Konstante	Bedeutung
DFHMSRST	Magnetcodeleser (MSR) zurückgesetzt. Alle Leuchtanzeigen und Summer inaktiv. MSR zur Eingabe verfügbar.
DFHMSCON	Transaktion für weitere Eingabe bereit. Grün und Gelb an; kurzes Akustiksignal ausgeben; Modus (Benutzermodus) IN PROCESS festgelegt.
DFHMSFIN	Eingabe abgeschlossen. Grün an; kurzes Akustiksignal ausgeben; Modus IN PROCESS zurückgesetzt.
DFHMSALR	Bedieneralert. Grün, Gelb und Rot an; langes Akustiksignal ausgeben; Modus IN PROCESS zurückgesetzt.
DFHMSALS	Bedieneralert. Grün, Gelb und Rot an; langes Akustiksignal ausgeben; Modus IN PROCESS festgelegt.
DFHMSIPY	Status IN PROCESS festgelegt. Gelb an.
DFHMSIPN	Status IN PROCESS zurückgesetzt.
DFHMSLKY	MSR-Operation unterdrückt. Gelb an.
DFHMSLKN	MSR-Eingabe zulässig. Grün an. Gelb an.
DFHMSAEY	Automatische Eingabe für MSR ein. Gelb an.
DFHMSAEN	Automatische Eingabe für MSR aus. Gelb an.
DFHMSLBN	Langes Summersignal unterdrückt. Gelb an.
DFHMSLBY	Langes Summersignal zugelassen. Gelb an.
DFHMSSBN	Kurzes Summersignal unterdrückt. Gelb an.
DFHMSSBY	Kurzes Summersignal zugelassen. Gelb an.
DFHMSNOP	Alle MSR-Einstellungen unverändert lassen.

STATE MASK

Wenn in der STATE MASK-Byte ein Bit aktiviert ist, wird der Status, den es darstellt, von der Einheit angenommen, sofern das entsprechende Bit auch im STATE VALUE-Byte aktiviert ist.

0 USER

Benutzermodus. Gelbes Licht einschalten, wenn das gleiche Bit in STATE VALUE aktiviert ist.

1 LOCK

Gesperrt/Entsperrt. Wenn gesperrt, wird die MSR-Eingabe unterdrückt.

2 AUTO

Automatische Eingabe ein/aus. Wenn eingeschaltet, bewirkt jeder Kartenlesevorgang durch den MSR eine Eingabeoperation (ENTER). Wenn ausgeschaltet, bewirkt nur eine sichere Karte eine Eingabeoperation.

3 Ai1S

Akustischen Alarm 1 unterdrücken.

4 Ai2S

Akustischen Alarm 2 unterdrücken.

STATE VALUE

Ändert den Status in ein oder aus, wenn das entsprechende Bit in STATE MASK auf ein gesetzt ist.

INDICATOR MASK

Führt eine ähnliche Funktion wie STATE MASK, jedoch für Anzeiger durch.

0

Licht 1 (Grün)

1

Licht 1 (Grün)

2

Licht 2 (Gelb)

3

Akustisches Signal 1 (langes Summersignal)

4

Akustisches Signal 2 (kurzes Summersignal)

INDICATOR VALUE

Führt eine ähnliche Funktion wie STATE VALUE aus.

AID-Zeichenkonstanten, DFHAID

Die AID-Standardliste DFHAID vereinfacht Tests des Inhalts des Felds EIBAID. In der folgenden Tabelle finden Sie den symbolischen Namen des AID-Zeichens (AID) und der entsprechenden 3270-Funktion.

Sie können eine Kopie der Liste abrufen, indem Sie DFHAID in Ihr Anwendungsprogramm kopieren. Für COBOL-Benutzer besteht die Liste aus einer Gruppe von 01-Anweisungen, die in den Arbeitsspeicherabschnitt (WORKING STORAGE SECTION) kopiert werden können. Für C-Benutzer besteht sie aus einer Reihe definierter Konstanten. Für PL/I-Benutzer besteht sie aus DECLARE-Anweisungen, die elementare Zeichenvariablen definieren.

In der folgenden Tabelle sind die Namen für AIDs aufgeführt.

Tabelle 1. Standardliste DFHAID	
Konstante	Bedeutung
DFHENTER	Eingabetaste (ENTER).

Tabelle 1. Standardliste DFHAID

(Forts.)

Konstante	Bedeutung
DFHCLEAR	Löschtaste (CLEAR).
DFHPA1– DFHPA3	Tasten PA1–PA3.
DFHPPF1– DFHPPF24	Tasten PF1–PF24.
DFHOPID	Operator-ID-Karte (OPERID) oder Magnetcodeleser (MSR).
DFHMSRE	Erweiterter Magnetcodeleser (Standard - MSR Extended).
DFHTRIG	Triggerfeld.
DFHPEN	Taste für Lichtstift (SELECTOR PEN) oder Cursorauswahl (CURSOR SELECT).
DFHCLRP ¹ auf Seite 847	Taste 'Partition löschen' (CLEAR PARTITION).
DFHSTRF ¹ auf Seite 847	Pseudo-AID für strukturiertes Feld.

Anmerkung:

1. DFHCLRP und DFHSTRF gelten nicht für minimalen Funktions-BMS.

Kapitel 9. BMS-Makros

Die Syntax der einzelnen BMS-Makros (BMS - Basic Mapping Support) wird definiert, wobei die verschiedenen Operanden und Optionen nach ihrer Eignung für minimale, standardmäßige und vollständige BMS unterteilt werden.

Geben Sie beim Codieren den Titel in Spalte 1, das Makro in Spalte 10 an. Fortsetzungszeilen müssen den Stern * in Spalte 72 enthalten und in Spalte 16 auf der nächsten Zeile fortgesetzt werden.

Weitere Informationen zu BMS finden Sie unter [Basic Mapping Support](#).

Maskengruppen-, Masken- und Felddefinition

Stellen Sie sicher, dass die Namen von Masken (Maps) und die Namen von Feldern innerhalb einer Maskengruppe (Mapset) bzw. innerhalb mehrerer Maskengruppen, die in dieselbe Anwendung kopiert werden, eindeutig sind. Eine Maske kann jedoch den gleichen Namen wie eine Maskengruppe haben.

Bevor CICS eine physische Maske laden kann, ist eine installierte Ressourcendefinition für das Maskenobjekt erforderlich. Sie können entweder die Definition mithilfe der automatischen Installation von Programmen bei der ersten Verwendung der Maskengruppe erstellen oder eine Maskengruppe in der CSD mithilfe der Ressourcendefinition DEFINE MAPSET definieren.

Sie assemblieren eine BMS-Maskengruppendefinition, um entweder eine symbolische Beschreibungsmaske oder eine physische Maske zu generieren. Die physische Maske ist ein strukturierter Datenbereich, der während der Ausführung zum Erstellen des Datenstroms für das Terminal verwendet wird. Die symbolische Maske ist eine Reihe von Datenstrukturen, die Sie bei der Kompilierung in Ihr Programm kopieren, sodass Sie auf die Felder in der Maske über ihre Namen verweisen können.

Programmierungsinformationen zum Benutzerprogramm für automatische Installation finden Sie in [Programm zur Steuerung der automatischen Installation von logischen Einheiten schreiben](#).

DFHMSD

Das Makro DFHMSD definiert eine Maskengruppe.

DFHMDI

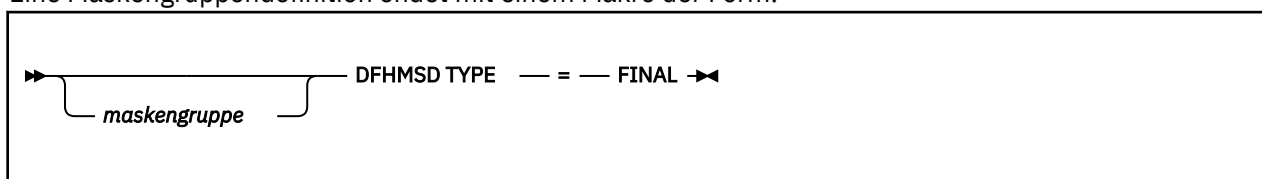
Das Makro DFHMDI definiert eine Maske (Map) innerhalb einer Maskengruppe (Mapset), die durch das vorherige Makro DFHMSD definiert wurde. Eine Maske enthält null oder mehr Felder.

DFHMDF

Das Makro DFHMDF definiert ein Feld in einer Maske (Map), die durch das vorherige Makro DFHMDI definiert wird.

Maskengruppendefinition beenden

Eine Maskengruppendefinition endet mit einem Makro der Form:



Der Wert "maskengruppe" ist optional. Wenn er jedoch verwendet wird, muss er mit dem Wert in dem Makro DFHMSD übereinstimmen, das die Maskengruppendefinition begonnen hat.

ADS-Deskriptor

Physische Masken, die von CICS Transaction Server for z/OS erzeugt werden, enthalten einen ADS-Deskriptor im Ausgabelademodul. Dieser wird bereitgestellt, um eine Interpretation der BMS-Anwendungs-

datenstruktur (ADS - Struktur, die vom Anwendungsprogramm für die Daten in SEND- und RECEIVE MAP-Anforderungen verwendet wird) zu ermöglichen, ohne dass Ihr Programm den relevanten DSECT oder das Copybook bei der Kompilierung enthalten muss.

Der ADS-Deskriptor enthält einen Header mit allgemeinen Informationen zu der Maske und einen Felddeskriptor für jedes Feld, das in der ADS (die jedem im Maskendefinitionsmakro benannten Feld entspricht) vorkommt.

Der ADS-Deskriptor wird für alle Masken generiert. Sie können die Option DSECT verwenden, um die Langform der ADS auszuwählen, in der alle Felder an 4-Byte-Grenzen ausgerichtet werden. Die lange Form des ADS ist für die 3270-Brücke erforderlich, wenn eine Schnittstelle zu IBM MQ verwendet wird.

Definition von Partitionsgruppen

Partitionen werden durch das Codieren der Makros DFHPSD (Partitionsgruppengruppendefinition) und DFHPDI (Partitionsdefinition) definiert. Jede Partitionsdefinition muss Teil einer Partitionsgruppengruppendefinition sein.

DFHPSD

Jede Partitionsgruppengruppendefinition enthält ein einzelnes Makro DFHPSD, auf das ein oder mehrere Makros DFHPDI sowie zum Schluss eine Partitionsgruppengruppendefinition TYPE=FINAL folgen.

Bevor CICS eine physische Maske laden kann, müssen Sie eine physische Maske unter Verwendung einer RDO-Transaktion mit dem Attribut MAPSET definieren.

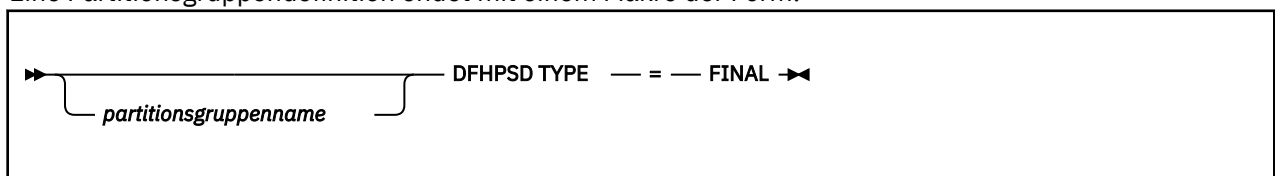
Alternativ zum Definieren von Masken mit RDO können Sie den Exit für die automatische Installation von Programmen verwenden, um die Definition bei der ersten Verwendung der Maskengruppe zu erstellen. (Programmierungsinformationen zum Benutzerprogramm für automatische Installation finden Sie in [Programm zur Steuerung der automatischen Installation von logischen Einheiten schreiben](#).)

DFHPDI

Eine Partitionsgruppe enthält eine oder mehr Partitionen. Jede Partition wird durch Codieren eines Partitionsdefinitionsmakros definiert.

Partitionsgruppengruppendefinition beenden

Eine Partitionsgruppengruppendefinition endet mit einem Makro der Form:



Der Partitionsgruppenname (sofern angegeben) muss mit dem Namen übereinstimmen, der im Makro DFHPSD angegeben wird, das die Partitionsgruppengruppendefinition gestartet hat.

Feldgruppen

Häufig muss ein Anzeigefeld für Ausgabedaten mehrere Unterfelder enthalten, die alle die gleichen Anzeigattribute verwenden und die möglicherweise einzeln geändert werden müssen.

Bei der Ausgabe können Unterfelder, die nicht durch das Programm geändert wurden, Standarddatenwerte aus der Ausgabemaske annehmen. Zum Beispiel kann eine Anzeige ein Datumsfeld eines Unterfelds vom Typ "day" (Tag), "month" (Monat) und "year" (Jahr) enthalten. Der Inhalt des Unterfelds für Jahr bleibt über einen relativ langen Zeitraum konstant und sein Wert kann problemlos aus einer Maske übernommen werden. Der Tageswert und der Monatswert müssen hingegen regelmäßig aktualisiert werden. Analog kann der Terminalbediener Daten separat in jedes Unterfeld eingeben.

Sie verwenden den Operanden GRPNAME, um eine Gruppe von Unterfeldern zu definieren, die kombiniert werden, um ein Feld zu erzeugen. Der Start der Gruppe wird durch ein Makro DFHMDF mit dem Operanden GRPNAME angegeben. Dieser Operand definiert das erste Unterfeld und gibt die Attribute und den Namen der Gruppe an. Darauf folgen weitere Makros DFHMDF für jedes der anderen Unterfelder. Jedes dieser Makros muss den Gruppennamen angeben, kann jedoch keine Attributwerte angeben. Die Definiti-

on der Gruppe wird durch ein Makro DFHMDF, das einen anderen Gruppennamen angibt, durch ein Makro, das keinen Gruppennamen oder durch ein Makro DFHMDI oder DFHMSD abgeschlossen.

Kurz gefasst sieht eine Gruppe von Feldern in einer Maske in der Maskendefinition wie folgt aus:

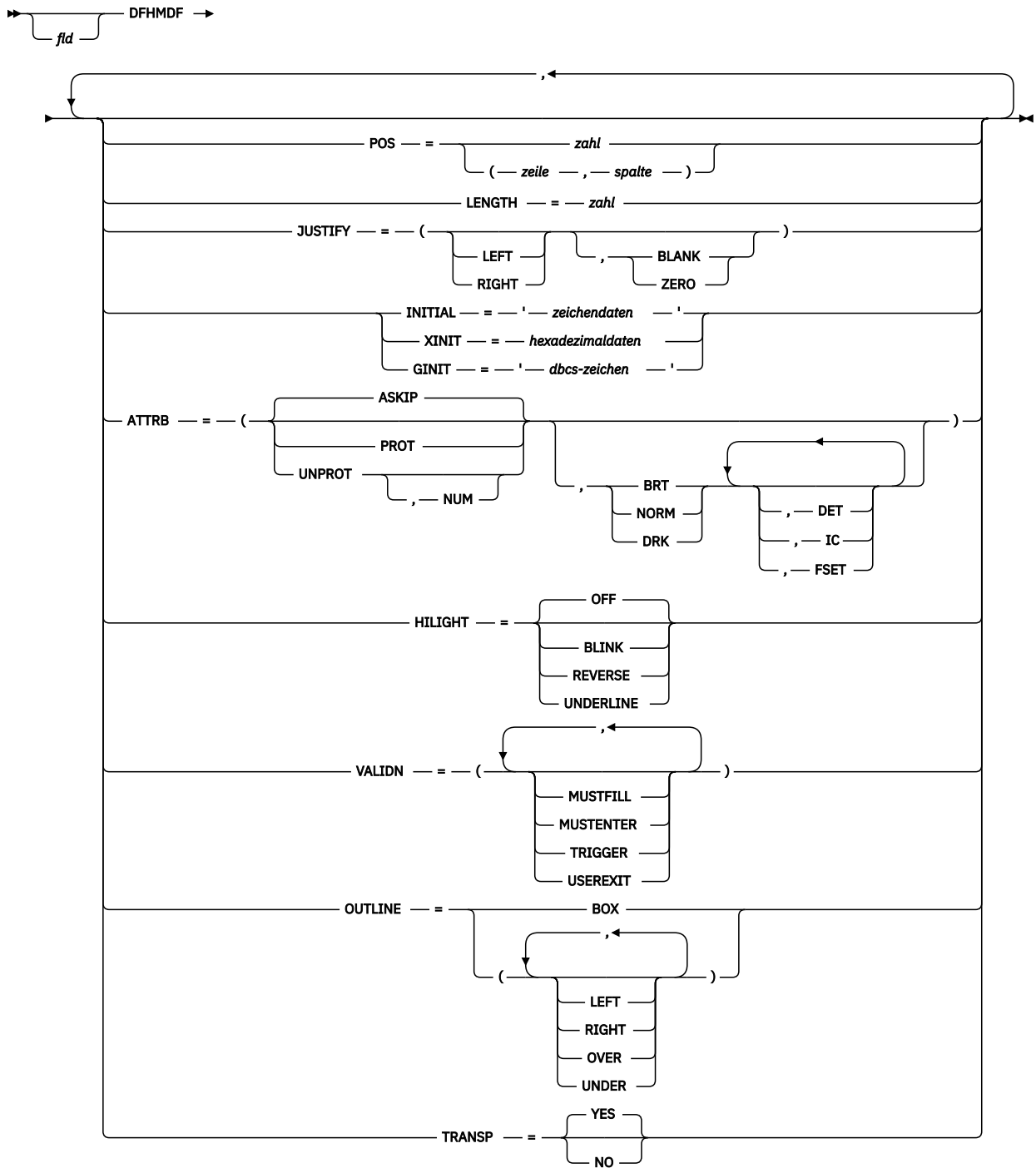
```
MAPSET DFHMSD . . .  
.  
MAP DFHMDI . . .  
.  
DD DFHMDF GRPNAME=DATE,POS=40,  
    LENGTH=2,ATTRB=. . .  
.  
MM DFHMDF GRPNAME=DATE,POS=46,  
    LENGTH=2  
.  
YY DFHMDF GRPNAME=DATE,POS=52,  
    LENGTH=2  
.  
FIELD DFHMDF LENGTH=5,COLOR=GREEN,. . .  
    DFHMSD TYPE=FINAL
```

Der Operand POS gibt die Position des Attributbytes des Felds an, auch wenn Unterfelder einer Gruppe (abgesehen vom ersten Unterfeld) keine Attribute haben. Wenn die Unterfelder zusammenhängend ohne Zwischenräume positioniert werden, muss der Operand POS des zweiten und der nachfolgenden Unterfelder die Position des letzten Zeichens des vorhergehenden Unterfelds angeben.

DFHMDF

Das Makro DFHMDF definiert ein Feld in einer Maske (Map), die durch das vorherige Makro DFHMDI definiert wird.

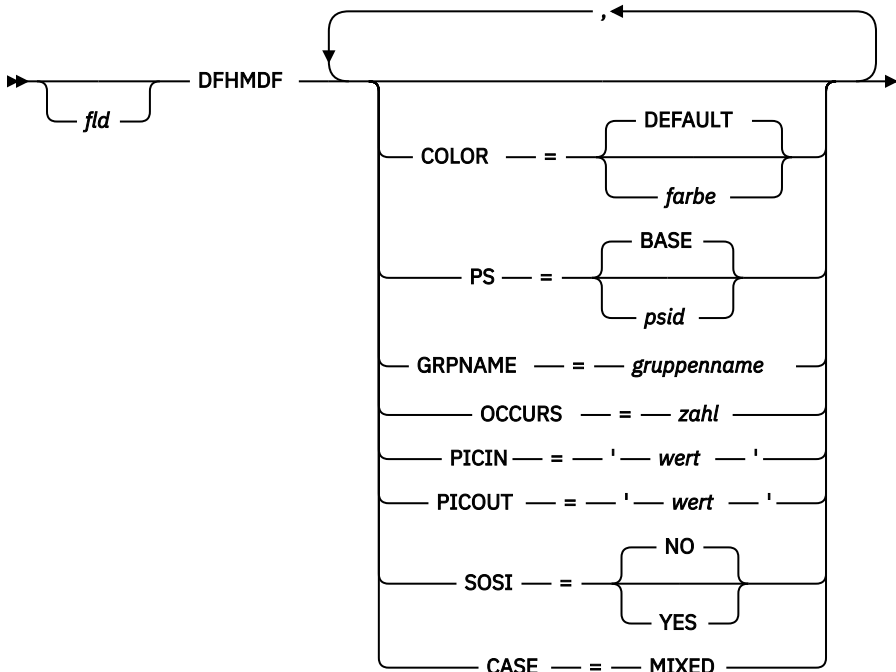
DFHMDf (Teil 1)



Anmerkungen:

¹ DBCS-Zeichen beginnen mit einem DBCS-Startzeichen X'0E' und enden mit einem DBCS-Endezeichen X'0F'.

DFHMDF (Teil 2)



Beschreibung

Eine Maske enthält null oder mehr Felder.

Der Wert **fld** ist der Name (1 – 30 Zeichen) des Felds. Lesen Sie die Informationen im Handbuch zu Ihrem Compiler, um sicherzustellen, dass keine anderen Einschränkungen für die Länge gelten.

Weitere Informationen zum Definieren von Feldnamen finden Sie unter [Zuordnung erstellen](#). Wenn der Wert **fld** nicht angegeben wird, können Programme nicht auf das Feld zugreifen, um seine Attribute oder seinen Inhalt zu ändern. Bei einer Ausgabezuordnung (Ausgabemaske) kann das Weglassen des Feldnamens geeignet sein, wenn der Operand INITIAL zur Angabe des Inhalts eines Felds verwendet wird. Wenn ein Feldname angegeben wird und die Maske, die das Feld enthält, in einer Zuordnungsoperation verwendet wird, überlagern Daten, die vom Benutzer angegeben werden, Daten, die durch die Initialisierung bereitgestellt werden (sofern nicht ausschließlich Standarddaten geschrieben werden).

Die Leistung von Eingabezuordnungsoperationen wird optimiert, wenn Makros DFHMDF in der numerischen Reihenfolge des Operanden POS angeordnet werden.

Sie können nicht mehr als 1023 benannte Felder für eine COBOL-, C- oder PL/I-Ein-/Ausgabemaske definieren.

Sie müssen sicherstellen, dass die Namen von Masken und die Namen von Feldern in einer Maskengruppe (bzw. in mehreren Maskengruppen, die in dasselbe Anwendungsprogramm kopiert werden) eindeutig sind. Eine Maske kann jedoch den gleichen Namen wie eine Maskengruppe haben.

Bevor CICS eine physische Maske laden kann, müssen Sie eine physische Maske unter Verwendung einer RDO-Transaktion mit dem Attribut MAPSET definieren.

Operanden

ATTRB

Dieser Operand gilt nur für 3270-Datenstromeinheiten. Für andere Einheiten wird er mit der einen Ausnahme ignoriert, dass die Angabe ATTRB=DRK für die logische Einheit mit 3270-SCS-Drucker berücksichtigt wird. Er wird außerdem (abgesehen von ATTRB=DRK) ignoriert, wenn die Option NLEOM im Befehl SEND MAP für die Übertragung an einen 3270-Drucker angegeben wird. Insbesondere darf

ATTRB=DRK nicht als Methode dazu verwendet werden, sichere Daten bei der Ausgabe auf Nicht-3270-Terminals oder Nicht-SCS-Druckerterminals zu schützen.

Wenn der Operand ATTRB in einer Gruppe von Feldern angegeben wird, muss er im ersten Feldeintrag angegeben werden. Eine Gruppe von Feldern erscheint für den 3270 wie ein Feld. Daher bezieht sich die Angabe von ATTRB auf alle Felder in einer Gruppe wie auf ein Feld und nicht wie auf einzelne Felder. Der Operand gibt einheitenabhängige Merkmale und Attribute an, wie zum Beispiel die Fähigkeit eines Felds, Daten zu empfangen, oder die Intensität, die bei der Ausgabe des Felds zu verwenden ist. Er könnte jedoch dazu verwendet werden, ein Eingabefeld zur sicheren Eingabe eines Kennworts über die Anzeige zur Unterdrückung der Anzeige einzurichten. Für Eingabemaskenfelder sind DET und NUM die einzigen gültigen Optionen. Alle anderen Optionen werden ignoriert.

ASKIP

Ist die Standardoption. Sie gibt an, dass Daten nicht in das Feld eingegeben werden können, und veranlasst, dass der Cursor das Feld überspringt.

BRT

Gibt an, dass eine Intensivanzeige des Felds erforderlich ist. Aufgrund der Zuweisungen von 3270-Attributzeichenbit ist ein mit BRT angegebenes Feld potenziell auch erkennbar. Damit das Feld von BMS als erkennbar eingestuft wird, muss auch die Option DET angegeben werden.

DET

Gibt an, dass das Feld potenziell erkennbar ('detectable') ist.

Das erste Zeichen eines erkennbaren 3270-Felds muss eines der folgenden sein:

? > & Leerzeichen

Wenn ? oder > verwendet wird, ist das Feld ein Auswahlfeld. Werden & oder ein Leerzeichen verwendet, ist das Feld ein Abruffeld. (Weitere Informationen zu erkennbaren (lichtstiftempfindlichen) Feldern finden Sie in der Einführung in das IBM 3270 Informationsanzeigesystem (*An Introduction to the IBM 3270 Information Display System*, IBM Form GA27-2739.)

Ein Feld, für das BRT angegeben wird, ist für das 3270 aufgrund der Zuweisungen von 3270-Attributzeichenbit potenziell erkennbar wird von BMS jedoch nur dann als solches eingestuft, wenn auch DET angegeben wird.

DET und DRK schließen sich gegenseitig aus.

Wenn für ein Feld in einer Maske mit MODE=IN die Option DET angegeben wird, wird für jedes Eingabefeld nur ein Datenbyte reserviert. Dieses Byte wird auf X'00' gesetzt und bleibt unverändert, wenn das Feld nicht ausgewählt wird. Wenn das Feld ausgewählt wird, wird das Byte auf X'FF' gesetzt.

Es werden keine anderen Daten bereitgestellt, auch wenn das Feld ein Auswahlfeld ist und die Eingabetaste gedrückt wurde.

Wenn die Daten in einem erkennbaren (lichtstiftempfindlichen) Feld erforderlich sind, müssen alle folgenden Bedingungen erfüllt werden:

1. Das Feld muss mit einem der folgenden Zeichen beginnen:

? > & Leerzeichen

und DET muss in der Ausgabemaske angegeben werden.

2. Die Eingabetaste (oder eine andere Abruftaste) muss nach der Auswahl des Felds gedrückt werden, obwohl die Eingabetaste für erkennbare Felder, die mit & oder einem Leerzeichen beginnen, nicht erforderlich ist.
3. DET darf für das Feld in der Eingabemaske (Eingabebezuordnung) nicht angegeben werden. DET muss jedoch in der Ausgabemaske (Ausgabebezuordnung) angegeben werden. Weitere Informationen zur BMS-Unterstützung für den Lichtstift finden Sie in der Veröffentlichung *CICS Application Programming Guide*.

DRK

Gibt an, dass für das Feld die Druckunterdrückung und die Anzeigeunterdrückung aktiv ist. DRK kann nicht angegeben werden, wenn DET angegeben wird.

FSET

Gibt an, dass das Tag für modifizierte Daten (MDT) für dieses Feld gesetzt sein muss, wenn das Feld an ein Terminal gesendet wird.

Die Angabe von FSET hat zur Folge, dass das 3270 das Feld so behandelt, als wäre es modifiziert worden. Bei einem nachfolgenden Lesevorgang vom Terminal wird dieses Feld gelesen, unabhängig davon, ob es modifiziert wurde. Das MDT bleibt gesetzt, bis das Feld ohne ATTRB=FSET erneut geschrieben wird oder bis eine Ausgabebezuordnungsanforderung veranlasst, dass das MDT zurückgesetzt wird.

Es kann ein beliebiger von zwei Sätzen von Standardwerten angewendet werden, wenn ein Feld, das auf einem 3270 angezeigt werden soll, definiert wird, aber nicht alle Parameter angegeben werden. Wenn keine ATTRB-Parameter angegeben werden, werden ASKIP und NORM angenommen. Wenn ein beliebiger Parameter angegeben wird, werden für dieses Feld UNPROT und NORM angenommen, sofern dies nicht durch einen angegebenen Parameter überschrieben wird.

IC

Gibt an, dass der Cursor an der ersten Position des Felds platziert werden soll. Das Attribut IC für das letzte Feld, für das es in einer Maske angegeben wird, ist dasjenige, das wirksam wird. Wenn es für keine Felder in einer Maske angegeben wird, ist die Standardposition null. Durch die Angabe von IC zusammen mit ASKIP oder PROT wird der Cursor in einem nicht eingabefähigen Feld platziert.

Diese Option kann durch die Option CURSOR des Befehls SEND MAP, der die Schreiboperation veranlasst, überschrieben werden.

NORM

Gibt an, dass die Intensität des Felds normal sein soll.

NUM

Stellt sicher, dass die Datenerfassungstastatur auf numerische Umschaltung für dieses Feld eingestellt wird, sofern der Bediener nicht die Alphaumschalttaste drückt, und verhindert die Eingabe nicht numerischer Daten, wenn die numerische Umschaltfunktion für die Tastatur (Keyboard Numeric Lock) installiert ist.

PROT

Gibt an, dass keine Daten über die Tastatur in das Feld eingegeben werden können.

Wenn Daten von einer Einheit auf eine andere Einheit, die an denselben 3270-Controller angeschlossen ist, kopiert werden sollen, darf die erste Position (Adresse 0) im Puffer der zu kopierenden Einheit kein Attributbyte für ein geschütztes Feld enthalten. Stellen Sie daher bei der Vorbereitung von Masken für 3270-Systeme sicher, dass die erste Maske einer beliebigen Seite kein geschütztes Feld enthält, das an Position 0 beginnt.

UNPROT

Gibt an, dass Daten über die Tastatur in das Feld eingegeben werden können.

CASE

Gibt an, dass das Feld Daten sowohl in Groß- als auch in Kleinbuchstaben enthält, die in Großbuchstaben umzusetzen sind, wenn die Terminaldefinition die Option für die Katakana-Unterstützung (KATAKANA(YES) in der RDO-TYPETERM-Definition) angibt.

Diese Option muss angegeben werden, wenn bekannt ist, dass ein Feld lateinische Buchstaben enthält, jedoch auf einer Katakana-Anzeige dargestellt werden kann. Sie darf nicht angegeben werden, wenn das Feld gültige Katakana-Zeichen enthalten kann.

COLOR

Gibt die individuelle Farbe oder die Standardfarbe für die Maskengruppe an (soweit zutreffend).

Die gültigen Farben sind blau (blue), rot (red), rosa (pink), grün (green), türkis (turquoise), gelb (yellow) und neutral (neutral).

Der Operand COLOR wird ignoriert, wenn das Terminal keine Farbe unterstützt, wie dies durch die RDO-Option COLOR angegeben wird.

GINIT

Gibt konstante Daten oder Standarddaten für ein Ausgabefeld an. GINIT wird zur Angabe von Daten in DBCS-Zeichenfolgen verwendet, die in DBCS-Startzeichen (X'0E') und DBCS-Endezeichen (X'0F') eingeschlossen werden müssen. Wenn GINIT angegeben wird, muss die Länge geradzahlig sein und sie stellt die Anzahl Byte in der Zeichenfolge dar (d. h., nicht die Anzahl der DBCS-Zeichen). Wenn eine grafischer Datentyp (PS=X'F8') verwendet wird und die Sprache mit COBOL2 (Enterprise COBOL oder VS COBOL II) angegeben ist, wird ein PIC G generiert. Es kann nur einer der Operanden GINIT, INITIAL oder XINIT angegeben werden.

GRPNAME

Ist der Name, der zum Generieren symbolischer Speicherdefinitionen sowie zum Kombinieren bestimmter Felder unter einem einzigen Gruppennamen verwendet wird. Für jedes Feld, das zu der Gruppe gehören soll, muss derselbe Gruppename angegeben werden. Die Länge des Namens kann bis zu 30 Zeichen betragen. Lesen Sie im Handbuch zu Ihrem Compiler nach, um sicherzustellen, dass keine anderen Einschränkungen für die Länge gelten.

Die Regeln für das Definieren von Gruppennamen sind mit denen zum Definieren von Feldnamen identisch. Weitere Informationen finden Sie unter [Zuordnung erstellen](#).

Wenn dieser Operand angegeben wird, kann der Operand OCCURS nicht angegeben werden.

Die Felder in einer Gruppe müssen aufeinander folgen. Es kann Lücken zwischen den Feldern geben, jedoch können keine anderen Felder von außerhalb der Gruppe vorhanden sein. Ein Feldname muss für jedes Feld angegeben werden, das zu der Gruppe gehört, und der Operand POS muss ebenfalls angegeben werden, um sicherzustellen, dass die Felder aufeinander folgen. Alle Makros DFHMDF, die Felder einer Gruppe definieren, müssen zusammen und in der richtigen Reihenfolge (aufsteigende numerische Reihenfolge des POS-Werts) angeordnet werden.

Beispiel: Die ersten 20 Spalten der ersten sechs Zeilen einer Maske können als Gruppe von sechs Feldern definiert werden, solange die verbleibenden Spalten in den ersten fünf Zeilen nicht als Felder definiert werden.

Der Operand ATTRB, der im ersten Feld der Gruppe angegeben wird, gilt für alle Felder in der Gruppe.

HILIGHT

Gibt das Standardhervorhebungsattribut für alle Felder in allen Masken einer Maskengruppe an.

OFF

Ist der Standardwert, der angibt, dass keine Hervorhebung verwendet wird.

BLINK

Gibt an, dass das Feld blinken muss.

REVERSE

Gibt an, dass das Zeichen oder Feld in Umkehranzeige dargestellt wird. Beispiel: Auf einem 3278 schwarze Zeichen auf grünem Hintergrund.

UNDERLINE

Gibt an, dass ein Feld unterstrichen wird.

Der Operand HILIGHT wird ignoriert, wenn das Terminal keine Hervorhebung unterstützt, wie dies durch die RDO-TYPETERM-Option HILIGHT(YES) angegeben wird.

INITIAL (oder XINIT)

Gibt konstante Daten oder Standarddaten für ein Ausgabefeld an. INITIAL wird zur Angabe von Daten in Zeichenform verwendet, während mit XINIT Daten in Hexadezimalform angegeben werden.

Für Felder mit dem Attribut DET müssen Anfangsdaten angegeben werden, die mit einem der folgenden Zeichen beginnen:

```
? > & Leerzeichen
```

Die Anzahl der Zeichen, die im Operanden INITIAL angegeben werden können, ist auf die Fortsetzungsbeschränkung des Assemblers, der verwendet werden soll, oder auf den Wert, der im Operanden LENGTH angegeben wird, (je nachdem, welcher Wert kleiner ist) beschränkt.

Hexadezimaldaten werden als gerade Zahl von Hexadezimalziffern geschrieben. Beispiel: XINIT=C1C2. Wenn die Anzahl gültiger Zeichen kleiner als die Feldlänge ist, werden die Daten auf der rechten Seite mit Leerzeichen aufgefüllt. Wenn zum Beispiel LENGTH=3 ist, ergibt XINIT=C1C2 ein Anfangsfeld mit 'AB'.

Wenn Hexadezimaldaten angegeben werden, die Zeilen- oder Formatsteuerzeichen entsprechen, sind die Ergebnisse unvorhersehbar. Der Operand XINIT muss daher mit Vorsicht verwendet werden. Es kann nur einer der Operanden GINIT, INITIAL oder XINIT angegeben werden.

JUSTIFY

Gibt die Feldausrichtungen für Eingabeoperationen an. Dieser Operand wird für VTAM-unterstützte Terminals 3600, 3650 und 3790 ignoriert, weil keine Eingabezuordnung verfügbar ist.

LEFT

Gibt an, dass Daten im Eingabefeld links ausgerichtet werden.

RIGHT

Gibt an, dass Daten im Eingabefeld rechts ausgerichtet werden.

BLANK

Gibt an, dass Leerzeichen an allen ungefüllten Positionen in einem Eingabefeld eingefügt werden sollen.

ZERO

Gibt an, dass Nullen an allen ungefüllten Positionen in einem Eingabefeld eingefügt werden sollen.

Die Optionen LEFT und RIGHT schließen sich gegenseitig aus, ebenso wie BLANK und ZERO. Wenn bestimmte Argumente angegeben werden, andere jedoch nicht, werden folgende Annahmen getroffen:

Angegeben	Angenommen
LEFT	BLANK
RIGHT	ZERO
BLANK	LEFT
ZERO	RIGHT

Wenn JUSTIFY nicht angegeben wird, jedoch das Attribut NUM angegeben wird, werden RIGHT und ZERO angenommen. Wenn JUSTIFY nicht angegeben wird, jedoch andere Attribute als NUM angegeben werden, werden LEFT und BLANK angenommen.

Anmerkung: Wenn ein Feld durch eine Ausgabezuordnung (Ausgabemaske) aus einer anderen Quelle initialisiert wird, überschreiben Daten, die als Eingabe getippt werden, nur die entsprechende Länge der vorhandenen Daten. Darüber hinaus vorhandene Daten verbleiben in dem Feld und könnten zu unerwarteten Interpretationen der neuen Daten führen.

LENGTH

Gibt die Länge (1–256 Byte) des Felds oder der Gruppe von Feldern an. Diese Länge ist die maximale Länge, die für die Eingabe von Anwendungsprogrammdateien in das Feld erforderlich ist. Sie schließt nicht den 1 Byte großen Attributanzeiger ein, der von CICS zur Verwendung in der nachfolgenden Verarbeitung an das Feld angehängt wird. Die Länge jedes einzelnen Unterfelds in einer Gruppe darf 256 Byte nicht überschreiten.

Im Allgemeinen kann LENGTH weggelassen werden, wenn PICIN oder PICOUT angegeben wird, sofern PICOUT kein COBOL-Bild definiert, das ein Währungssymbol enthält, das ein Währungssymbol einer Länge größer als 1 ersetzt. Andernfalls ist LENGTH erforderlich. Sie können die Länge null nur dann angeben, wenn Sie die Bezeichnung (Feldname) im Makro DFHMDF weglassen. Dies bedeutet, dass das Feld kein Teil der Anwendungsdatenstruktur ist und das Anwendungsprogramm die Attribute des Felds nicht ändern kann. Sie können ein Feld mit der Länge null verwenden, um ein Eingabefeld in einer Maske zu begrenzen.

Die Maskenabmessungen, die im Operanden SIZE des Makros DFHMDI angegeben werden, das eine Maske definiert, können kleiner als die tatsächliche Seitengröße oder Anzeigegröße sein, die für das Terminal definiert ist.

Wenn die Angabe von LENGTH in einem Makro DFHMDF zur Folge hat, dass die maskendefinierte Grenze auf derselben Zeile überschritten wird, wird das Feld in der Ausgabeanzeige durch Umbruch fortgesetzt.

OCCURS

Gibt an, dass die angegebene Anzahl von Einträgen für das Feld in einer Maske zu generieren ist und dass die Maskendefinition so zu generieren ist, dass die Felder als Einträge in einer Matrix oder einer Elementgruppe adressierbar sind. Dadurch lassen sich mehrere Datenfelder durch denselben Namen (indexiert) adressieren, ohne dass ein eindeutiger Name für jedes Feld generiert wird.

Die Operanden OCCURS und GRPNAME schließen sich gegenseitig aus. Das heißt, dass OCCURS nicht verwendet werden kann, wenn Felder unter einem Gruppennamen definiert wurden. Wenn dieser Operand weggelassen wird, wird der Wert OCCURS=1 angenommen.

OUTLINE

Ermöglicht, dass Linien über, unter, links oder rechts von einem Feld eingeschlossen werden. Mithilfe einer beliebigen Kombination dieser Linien können Sie Rahmen um Felder oder Gruppen von Feldern erstellen.

PICIN (nur COBOL und PL/I)

Gibt ein Bild an, das auf ein Eingabefeld in einer IN- oder INOUT-Maske angewendet wird. Dieses Bild dient als Bearbeitungsspezifikation, die an das Anwendungsprogramm übergeben wird, sodass der Benutzer die Bearbeitungsfunktionen von COBOL oder PL/I nutzen kann. BMS prüft, ob die angegebenen Zeichen gültige Bildspezifikationen für die Sprache der Maske sind.

Die Gültigkeit der Eingabedaten oder die höhere Programmiersprache wird von BMS jedoch nicht geprüft, wenn die Maske verwendet wird, sodass alle gewünschten Prüfungen durch das Anwendungsprogramm erfolgen müssen. Die Länge der Daten, die dem Wert "wert" zugeordnet werden, muss mit der Länge übereinstimmen, die im Operanden LENGTH angegeben wird, wenn LENGTH angegeben wird. Wenn sowohl PICIN als auch PICOUT verwendet werden, wird eine Fehlernachricht generiert, wenn ihre berechneten Längen nicht übereinstimmen. Die kürzere der beiden Längen wird verwendet. Wenn PICIN oder PICOUT nicht für die Felddefinition codiert wird, wird automatisch eine Zeichendefinition des Felds generiert, und zwar unabhängig von anderen Operanden, die codiert werden, wie zum Beispiel ATTRB=NUM.

Nehmen Sie zum Beispiel an, dass die folgende Maskendefinition zur Referenz durch ein COBOL-Anwendungsprogramm erstellt wird:

MAPX	DFHMSD	TYPE=DSECT, LANG=COBOL, MODE=INOUT
MAP	DFHMDI	LINE=1,COLUMN=1, SIZE=(1,80)
F1	DFHMDF	POS=0,LENGTH=30
F2	DFHMDF	POS=40,LENGTH=10, PICOUT='\$\$\$,\$\$0.00'
F3	DFHMDF	POS=60,LENGTH=6, PICIN='9999V99', PICOUT='ZZ9.99'
	DFHMSD	TYPE=FINAL

Dadurch wird der folgende DSECT generiert:

01	MAPI.	
02	F1L	PIC S9(4) COMP.
02	F1A	PIC X.
02	FILLER	REDEFINES F1A.
03	F1F	PIC X.
02	F1I	PIC X(30).
02	FILLER	PIC X.
02	F2L	PIC S9(4) COMP.
02	F2A	PIC X.
02	FILLER	REDEFINES F2A.
03	F2F	PIC X.

```

02 F2I      PIC X(10).
02 FILLER   PIC X.
02 F3L      PIC S9(4) COMP.
02 F3A      PIC X.
02 FILLER   REDEFINES F3A.
03 F3F      PIC X.
02 F3I      PIC 9999V99.
02 FILLER   PIC X.

```

```

01 MAPO REDEFINES MAPI.
02 FILLER   PIC X(3).
02 F10      PIC X(30).
02 FILLER   PIC X.
02 FILLER   PIC X(3).
02 F20      PIC $$$,$$0.00.
02 FILLER   PIC X.
02 FILLER   PIC X(3).
02 F30      PIC ZZ9.99.
02 FILLER   PIC X.

```

Gültige Bildwerte für COBOL-Eingabemasken sind folgende:

A P S V X 9 / und (

Gültige Bildwerte für PL/I-Eingabemasken sind folgende:

A B E F G H I K M P R S T V
X Y und Z
1 2 3 6 7 8 9 / + - , . *
\$ und (

Im Referenzhandbuch für die jeweilige Programmiersprache finden Sie Informationen zur korrekten Syntax des Attributs PICTURE.

Anmerkung: PL/I unterstützt mehrere Währungssymbole und aus mehreren Zeichen bestehende Währungssymbole in PICTURE-Spezifikationen.

Das Standardwährungsbildsymbol ist das Dollarzeichen (\$), das das nationale Währungssymbol darstellt, wie zum Beispiel den Dollar (\$), das Pfund (£) oder den Yen (¥).

Das Standardwährungsbildsymbol kann durch eine Währungszeichenfolge ersetzt werden, die von einem Kleiner-als-Zeichen (<) und einem Größer-als-Zeichen (>) umschlossen wird. Beispiel:

```

DECLARE
  USPRICE PICTURE '$99.99',
  UKPRICE PICTURE '<£>99.99',
  EUPRICE PICTURE '<EUR>99.99';

```

PICOUT (nur COBOL und PL/I)

Ist PICIN ähnlich, jedoch wird ein Bild generiert, das auf ein Ausgabefeld in der OUT- oder INOUT-Maske angewendet werden sollen.

Gültige Bildwerte für COBOL-Ausgabemasken sind folgende:

A B E P S V X Z 0 9 , . + - \$
CR DB / und (

Gültige Bildwerte für PL/I-Ausgabemasken sind folgende:

A B E F G H I K M P R S T V
X Y und Z
1 2 3 6 7 8 9 / + - , . * \$
CR DB und (

Im Referenzhandbuch für die jeweilige Programmiersprache finden Sie Informationen zur korrekten Syntax des Attributs PICTURE.

Anmerkung: PL/I unterstützt mehrere Währungssymbole und aus mehreren Zeichen bestehende Währungssymbole in PICTURE-Spezifikationen.

Das Standardwährungsbildsymbol ist das Dollarzeichen (\$), das das nationale Währungssymbol darstellt, wie zum Beispiel den Dollar (\$), das Pfund (£) oder den Yen (¥).

Das Standardwährungsbildsymbol kann durch eine Währungszeichenfolge ersetzt werden, die von einem Kleiner-als-Zeichen (<) und einem Größer-als-Zeichen (>) umschlossen wird. Beispiel:

```
DECLARE
  USPRICE PICTURE '$99.99',
  UKPRICE PICTURE '<£>99.99',
  EUPRICE PICTURE '<EUR>99.99';
```

Anmerkung: COBOL unterstützt mehrere Währungssymbole und aus mehreren Zeichen bestehende Währungssymbole in PICTURE-Spezifikationen.

Das Standardwährungsbildsymbol ist das Dollarzeichen (\$), das das nationale Währungssymbol darstellt, wie zum Beispiel den Dollar (\$), das Pfund (£) oder den Yen (¥).

Das Standardwährungsbildsymbol kann durch ein anderes Währungsbildsymbol ersetzt werden, das in der Klausel SPECIAL NAMES definiert wird. Das Währungssymbol, das durch das Bildsymbol dargestellt wird, wird in derselben Klausel definiert. Beispiel:

```
SPECIAL NAMES.
CURRENCY SIGN IS '$' WITH PICTURE SYMBOL '$'.
CURRENCY SIGN IS '£' WITH PICTURE SYMBOL '£'.
CURRENCY SIGN IS 'EUR' WITH PICTURE SYMBOL '#'.

WORKING STORAGE SECTION.
01 USPRICE PIC $99.99.
01 UKPRICE PIC £99.99.
01 ECPRICE PIC #99.99.
```

Der Operand LENGTH muss angegeben werden, wenn PICOUT ein COBOL-Bild mit einem Währungssymbol angibt, das ein Währungssymbol mit einer Länge größer 1 ersetzt.

POS

Gibt die Position eines Felds an. Dieser Operand gibt die individuell adressierbare Zeichenposition in einer Maske, an der das Attributbyte positioniert ist, das dem Feld vorausgeht.

zahl

Gibt die Distanz (relativ zu null) vom Anfang der Maske an, die definiert wird.

(zeile,spalte)

Gibt Zeilen und Spalten (relativ zu 1) in der Maske an, die definiert wird.

Die Position von Daten im Ausgabemedium hängt außerdem von DFHMDI-Operanden ab.

Die erste Position eines Felds ist für ein Attributbyte reserviert. Wenn Daten für die Eingabezuordnung über Nicht-3270-Einheiten eingegeben werden, müssen die Eingabedaten Platz für dieses Attributbyte lassen. Eingabedaten dürfen nicht in Spalte 1 beginnen, sondern können in Spalte 2 beginnen.

Der Operand POS enthält immer die Angabe der ersten Position in einem Feld, die bei der Kommunikation mit der 3270 normalerweise das Attributbyte enthält. Für das zweite und nachfolgende Felder einer Gruppe verweist der Operand POS auf eine angenommene Attributbyteposition vor dem Beginn der Daten, auch wenn kein tatsächliches Attributbyte erforderlich ist. Wenn die Felder direkt aufeinander folgen, verweist der Operand POS auf die letzte Zeichenposition im vorherigen Feld in der Gruppe.

Wenn eine Positionsnummer angegeben wird, die die letzte Zeichenposition in der 3270 darstellt, gelten zwei Sonderregeln:

- ATTRIB=IC darf nicht codiert werden. Der Cursor kann mithilfe der Option CURSOR eines Befehls SEND MAP, SEND CONTROL oder SEND TEXT auf die Position null gesetzt werden.

- Wenn das Feld in einer Ausgabezuordnungsoperation mit MAP=DATAONLY im Befehl SEND MAP verwendet werden soll, muss durch das Anwendungsprogramm ein Attributbyte in der symbolischen Maskendatenstruktur angegeben werden.

PS

Gibt an, dass programmierte Symbole verwendet werden sollen. Diese Angabe überschreibt jeden Operanden PS, der durch das Makro DFHMDI oder das Makro DFHMSD festgelegt wird.

BASE

Ist der Standard und gibt an, dass der Basissymbolsatz verwendet werden soll.

psid

Gibt ein einzelnes EBCDIC-Zeichen oder einen Hexadezimalcode der Form X'nn' an, der den zu verwendenden Satz programmierter Symbole bezeichnet.

Der Operand PS wird ignoriert, wenn das Terminal keine programmierten Symbole unterstützt, wie dies durch PROGSYMBOLS(YES) in der RDO-TYPETERM-Definition angegeben wird.

SOSI

Gibt an, dass das Feld eine Mischung aus EBCDIC- und DBCS-Daten enthalten kann. Die DBCS-Unterfelder in einem EBCDIC-Feld werden durch ein DBCS-Startzeichen (SO - Shift-out) und ein DBCS-Endezeichen (SI - Shift-in) begrenzt. Das Start- und das Endezeichen nehmen beide eine einzelne Anzeigeposition ein (die normalerweise als Leerzeichen angezeigt wird). Sie können in einer Ausgabe in ein beliebiges Nicht-DBCS-Feld eingeschlossen werden, sofern sie paarig korrekt angegeben werden. Der Terminalbenutzer kann sie eingehend übertragen, wenn sie bereits im Feld vorhanden sind. Er kann sich einem EBCDIC-Feld jedoch nur dann hinzufügen, wenn das Feld das Attribut SOSI besitzt.

TRANSP

Bestimmt, ob der Hintergrund eines alphanumerischen Felds transparent oder nicht transparent (opak) ist, das heißt, ob ein darunter liegender (grafischer) Darstellungsbereich zwischen den Zeichen sichtbar ist.

VALIDN

Gibt Folgendes an:

- Dass eine Prüfung (Validierung) auf einem 8775-Terminal verwendet wird.
- Dass dieses Feld von globalen BMS-Benutzerexits verarbeitet werden kann.

Diese Angabe überschreibt jeden Operanden VALIDN im Makro DFHMDI oder im Makro DFHMSD.

MUSTFILL

Gibt an, dass das Feld vollständig mit Daten gefüllt werden muss. Ein Versuch, den Cursor von dem Feld wegzubewegen, bevor es gefüllt wurde, oder Daten aus einem unvollständigen Feld zu übertragen, löst die Bedingung INHIBIT INPUT aus.

MUSTENTER

Gibt an, dass Daten in das Feld eingegeben werden müssen, es jedoch nicht völlig ausfüllen müssen. Ein Versuch, den Cursor von einem leeren Feld wegzubewegen, löst die Bedingung INHIBIT INPUT aus.

TRIGGER

Gibt an, dass dieses Feld ein Triggerfeld (Auslöserfeld) ist. Triggerfelder werden in [Unterstützung für Triggerfeld](#) beschrieben.

USEREXIT

Gibt an, dass dieses Feld durch die globalen BMS-Benutzerexits XBMIN und XBMOU verarbeitet werden soll, wenn dieses Feld in einem 3270-Datenstrom empfangen oder übertragen wird, sofern der entsprechende Exit aktiviert ist.

Die Operanden MUSTFILL, MUSTENTER und TRIGGER sind nur für Terminals gültig, die das erweiterte Attribut für Feldprüfung unterstützen. Andernfalls werden sie ignoriert. Die USEREXIT-Spezifikation gilt für alle 3270-Einheiten.

Anmerkung: Die USEREXIT-Spezifikation ist nicht mit dem erweiterten Attribut für Feldprüfung verbunden, wie dies in der 3270-Datenstromarchitektur definiert ist.

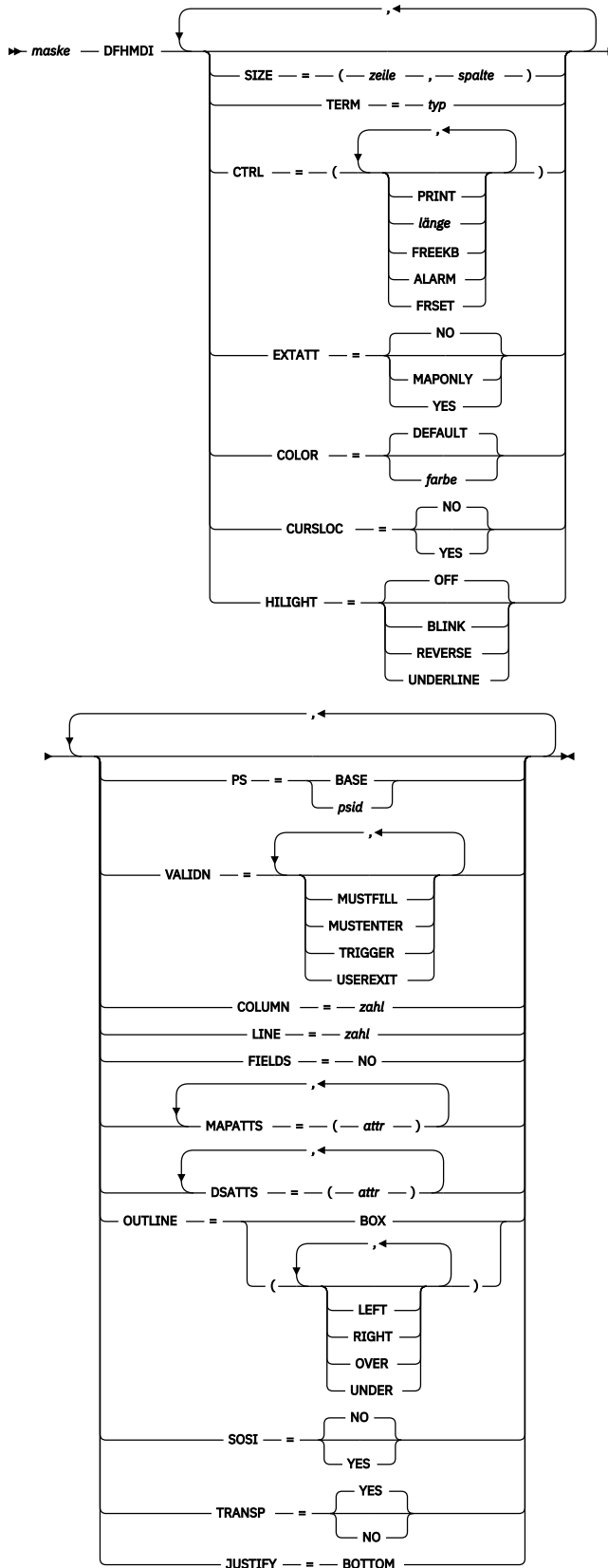
XINIT

Siehe INITIAL weiter oben in der Liste. Es kann nur einer der Operanden GINIT, INITIAL oder XINIT angegeben werden.

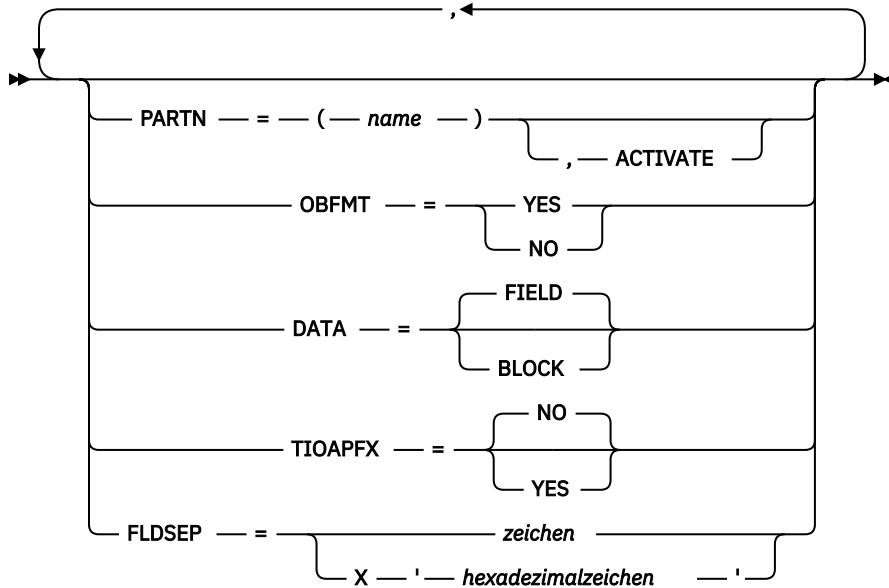
DFHMDI

Das Makro DFHMDI definiert eine Maske (Map) innerhalb einer Maskengruppe (Mapset), die durch ein vorheriges Makro DFHMSD definiert wurde.

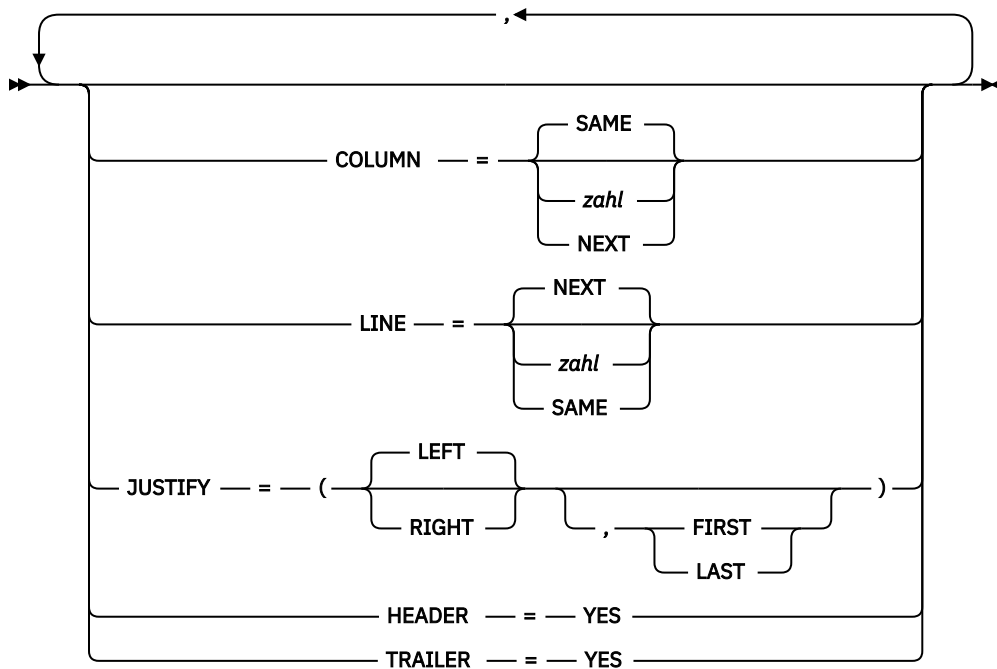
Minimaler BMS für DFHMDI



Standard-BMS für DFHMDI



Vollständiger BMS für DFHMDI



Eine Maske enthält null oder mehr Felder.

Der Wert "maske" ist der Name (1-7 Zeichen) der Maske.

Operanden

COLOR

Gibt die individuelle Farbe oder die Standardfarbe für die Maskengruppe an (soweit zutreffend). Dies wird durch den Operanden COLOR des Makros DFHMDF überschrieben.

Die gültigen Farben sind blau (blue), rot (red), rosa (pink), grün (green), türkis (turquoise), gelb (yellow) und neutral (neutral).

Der Operand COLOR wird ignoriert, wenn das Terminal keine Farbe unterstützt, wie dies durch die Option COLOR(YES) der RDO-TYPETERM-Definition angegeben wird.

COLUMN

Gibt die Spalte in einer Zeile an, in der die Maske positioniert werden soll, das heißt, dieser Operand richtet den linken oder rechten Maskenrand ein. Der Operand JUSTIFY im Makro DFHMDI steuert, ob die Randauswahl für die Maske und die Seite sowie die Spaltenzählung von der linken oder rechten Seite der Anzeige aus erfolgen soll. Die Spalten zwischen dem angegebenen Maskenrand und dem Seitenrand sind für eine nachfolgende Verwendung auf der Seite für Zeilen, die in der Maske enthalten sind, nicht verfügbar.

NUMBER

Gibt die Spalte vom linken oder rechten Seitenrand an, in der der linke oder rechte Maskenrand positioniert werden soll.

NEXT

Gibt an, dass der linke oder rechte Maskenrand in der nächsten verfügbaren Spalte links oder rechts von der aktuellen Zeile positioniert werden soll.

SAME

Gibt an, dass der linke oder rechte Maskenrand in derselben Spalte positioniert werden soll wie die letzte verwendete Maske, die kein Header oder Trailer ist, die COLUMN=zahl angibt, und dieselben JUSTIFY-Operanden wie in diesem Makro verwendet werden sollen.

Für Eingabeoperationen wird die Maske ganz links oder rechts positioniert, je nachdem, ob JUSTIFY=LEFT oder JUSTIFY=RIGHT angegeben wurde.

CTRL

Definiert Merkmale von IBM 3270-Terminals. Die Verwendung von Steueroptionen im Befehl SEND MAP überschreibt alle Steueroptionen im Makro DFHMDI, das wiederum alle Steueroptionen im Makro DFHMSD überschreibt.

Wenn CTRL mit der kumulativen BMS-Seitenverarbeitung (d. h. die Option ACCUM wird in BMS-Befehlen SEND MAP verwendet) verwendet wird, muss CTRL in der letzten (oder einzigen) Maske einer Seite angegeben werden, sofern dies nicht durch die Optionen ALARM, FREEKB usw. im Befehl SEND MAP oder im kumulierten Befehl SEND CONTROL überschrieben wird.

PRINT

Muss angegeben werden, wenn der Drucker gestartet werden soll. Wenn nicht angegeben, werden die Daten an den Druckerpuffer gesendet, jedoch nicht gedruckt. Dieser Operand wird ignoriert, wenn die Maskengruppe mit 3270-Anzeigen ohne Druckeradapterfunktion verwendet wird.

LENGTH

Gibt die Zeilenlänge auf dem Drucker an. Die Länge kann als L40, L64, L80 oder HONEOM angegeben werden. Die Angaben L40, L64 und L80 erzwingen eine neue Zeile nach 40, 64 bzw. 80 Zeichen. Bei Angabe von HONEOM wird die Standardzeilenlänge des Druckers verwendet. Wenn diese Option nicht angegeben wird, legt BMS die Zeilenlänge entsprechend der TCT-Seitengröße fest.

FREEKB

Veranlasst, dass die Tastatur entsperrt wird, nachdem die Maske geschrieben wurde. Wenn FREEKB nicht angegeben wird, bleibt die Tastatur gesperrt. Die Dateneingabe über die Tastatur wird unterbunden, bis dieser Status geändert wird.

ALARM

Aktiviert das akustische 3270-Signal. Für Nicht 3270-VTAM-Terminals setzt dieser Operand das Alarmflag im Funktionsverwaltungsheader (Function Management Header - FMH). (Diese Funktion wird von interaktiven Einheiten und logischen Einheiten mit Stapelverarbeitung nicht unterstützt).

FRSET

Gibt an, dass die Tags für modifizierte Daten (MDTs) aller Felder, die sich momentan im 3270-Puffer befinden, auf die Bedingung 'Nicht geändert' zurückgesetzt werden sollen (Feldzurückset-

zung), bevor Maskendaten in den Puffer geschrieben werden. Dadurch kann das Makro DFHMDF mit dem Operanden ATTRB den finalen Status beliebiger Felder steuern, die als Antwort auf einen BMS-Befehl geschrieben oder erneut geschrieben werden.

Anmerkung: CTRL kann nicht in den Makros DFHMDI und DFHMSD in derselben Maskengruppe angegeben werden.

CURSLOC

Gibt an, dass BMS für alle Operationen RECEIVE MAP mit dieser Maske auf 3270-Terminals ein Flag in dem Anwendungsdatenstrukturelement für das Feld setzt, in dem sich der Cursor befindet.

Das Flag kann von DFHBMCUR getestet werden (siehe Copybook DFHBMSCA in [Kapitel 8, „BMS-spezifische Konstanten“](#), auf Seite 841).

Das Flag kann wie folgt getestet werden (COBOL-Beispiel):

```
(DFHBMSCA)
...
02 DFHBMEOF PIC X VALUE X'80'.
02 DFHBMCUR PIC X VALUE X'02'.
02 DFHBMEC PIC X VALUE X'82'.
02 DFHBMFLG PIC X.
88 DFHERASE VALUES ARE X'80', X'82'.
88 DFHCURSR VALUES ARE X'02', X'82'.
MOVE FLD1F TO DFHBMFLG.
IF DFHERASE THEN ...
ELSE ...
IF DFHCURSR THEN ...
ELSE ...
```

Anmerkung:

1. Wenn CURSLOC=YES für die MAP-Definitionen angegeben wird und keine Daten für ein beliebiges Feld der Anwendungsdatenstruktur vorhanden sind, aber der Cursor in einem Feld liegt, das der Anwendungsdatenstruktur bekannt ist, setzt BMS das Cursor-Flag für das entsprechende Feld, jedoch sind die Daten für alle Felder in der Anwendungsdatenstruktur null und die Bedingung MAP-FAIL tritt nicht auf. Der in der Maske nicht zugeordnete Datenstrom ist für das Anwendungsprogramm nur verfügbar, wenn es sich um eine Anforderung RECEIVE DATA FROM handelt.
2. Eine gültige CURSLOC-Definition in DFHMDI überschreibt die Definition in DFHMSD.

DATA

Gibt das Format der Daten an.

FIELD

Gibt an, dass die Daten als fortlaufende Felder übergeben werden, wobei jedes Feld das folgende Format hat:

LL	A	Datenfeld
----	---	-----------

“LL” sind zwei Byte, die die Länge der Daten als Eingabe vom Terminal angeben (werden bei der Ausgabeverarbeitung ignoriert). “A” ist ein Byte, in dem der Programmierer ein Attribut platzieren kann, um das in der Maskengruppe angegebene Attribut zu überschreiben, das zur Verarbeitung dieser Daten verwendet wird (siehe Copybook DFHBMSCA in [Kapitel 8, „BMS-spezifische Konstanten“](#), auf Seite 841).

BLOCK

Gibt an, dass die Daten als fortlaufender Datenstrom im folgenden Format übergeben wird:

A	Datenfeld	Leerzeichen
---	-----------	-------------

Dieser Datenstrom wird als Zeilensegmente der Länge verarbeitet, die in der Maske angegeben wird, die zur Verarbeitung der Daten verwendet wird. Die Daten liegen in der Form vor, in der sie auf dem Terminal dargestellt werden. Das heißt, sie enthalten Datenfelder und dazwischen eingefügte Leerzeichen, die den Leerzeichen entsprechen, die zwischen den Feldern in der Ausgabe an-

gezeigt werden sollen. Sie können DSATTS=YES nicht verwenden, wenn Sie DATA=BLOCK angeben.

Blockdaten werden unter [Blockdaten](#) eingehender erläutert.

DSATTS

Gibt die Attributtypen an, die in die symbolische Beschreibungsmaske eingeschlossen werden sollen. Diese Typen können einer oder mehrere der folgenden sein: COLOR, HILIGHT, OUTLINE, PS, SOSI, TRANSP und VALIDN. Jeder Typ, der in DSATTS eingeschlossen wird, sollte auch in MAPATTS eingeschlossen werden.

EXTATT

Dieser Operand wird aus Gründen der Kompatibilität mit früheren Releases unterstützt. Jedes der erweiterten Attribute kann einzeln definiert werden. Für neue Masken müssen stattdessen die Operanden DSATTS und MAPATTS verwendet werden.

NO

Ist äquivalent dazu, den Operanden DSATTS oder den Operanden MAPATTS nicht anzugeben.

YES

Ist äquivalent zu folgender Angabe:

```
MAPATTS=(COLOR,HILIGHT,PS,VALIDN)
DSATTS=(COLOR,HILIGHT,PS,VALIDN)
```

MAPONLY

Ist äquivalent zu folgender Angabe:

```
MAPATTS=(COLOR,HILIGHT,PS,VALIDN)
```

FIELDS

Gibt an, ob die Maske Felder enthält. Wenn Sie FIELDS=NO angeben, erstellen Sie eine Nullmaske, die eine "Lücke" in der BMS-Ansicht der Anzeige definiert. BMS kann den Inhalt solcher Lücken nach der Erstellung durch Senden einer Nullmaske nicht ändern.

FLDSEP

Gibt die Feldtrennzeichensequenz (1-4 Zeichen) für Eingaben über Nicht-3270-Einheiten an. Eingaben über Nicht-3270-Einheiten können als einzelne Zeichenfolge von Daten mit der Feldtrennzeichensequenz als Feldbegrenzung eingegeben werden. Die Daten zwischen den Feldtrennzeichen werden in der Reihenfolge in die Eingabefelder der Maske versetzt.

HEADER

Ermöglicht, dass die Maske bei der Seitenerstellung verwendet werden kann, ohne die Bedingung OVERFLOW zu beenden. Dieser Operand kann für mehr als eine Maske in einer Maskengruppe angegeben werden.

HILIGHT

Gibt das Standardhervorhebungsattribut für alle Felder in allen Masken einer Maskengruppe an. Dies wird durch den Operanden HILIGHT des Makros DFHMDF überschrieben.

OFF

Ist der Standardwert, der angibt, dass keine Hervorhebung verwendet wird.

BLINK

Gibt an, dass das Feld blinken muss.

REVERSE

Gibt an, dass das Zeichen oder Feld in Umkehranzeige dargestellt wird. Beispiel: Auf einem 3278 schwarze Zeichen auf grünem Hintergrund.

UNDERLINE

Gibt an, dass ein Feld unterstrichen wird.

Der Operand HILIGHT wird ignoriert, wenn das Terminal keine Hervorhebung unterstützt, wie dies durch die Option HILIGHT(YES) in der RDO-TYPETERM-Definition angegeben wird.

JUSTIFY

Gibt die Position der Maske auf der Seite an.

LEFT

Gibt an, dass die Maske beginnend von der angegebenen Spalte vom linken Rand der angegebenen Zeile positioniert werden soll.

RIGHT

Gibt an, dass die Maske beginnend von der angegebenen Spalte vom rechten Rand der angegebenen Zeile positioniert werden soll.

FIRST

Gibt an, dass die Maske als erste Maske auf einer neuen Seite positioniert werden soll. Jede teilweise formatierte Seite aus vorangehenden BMS-Befehlen wird als vollständig betrachtet. Dieser Operand kann nur für eine Maske pro Seite angegeben werden.

LAST

Gibt an, dass die Maske im Fußbereich der aktuellen Seite positioniert werden soll. Dieser Operand kann nur für mehrere Masken angegeben werden, die auf einer Seite platziert werden sollen. Allerdings müssen sich andere Masken als die erste Maske, für die dieser Operand angegeben wird, horizontal positionieren lassen, ohne dass die Verwendung weiterer Zeilen erforderlich wird.

BOTTOM

Hat für einen Befehl SEND MAP ACCUM denselben Effekt wie LAST. Für einen Befehl SEND MAP (ohne ACCUM) und einen Befehl RECEIVE MAP positioniert die Angabe JUSTIFY=BOTTOM die Maske im Fußbereich der Anzeige, wenn die Anzahl der Zeilen in der Maske im Operanden SIZE angegeben wird. Trailermasken in der Maskengruppe werden dabei nicht berücksichtigt. Die Angabe JUSTIFY=BOTTOM ist äquivalent zu

```
LINE=(anzeigentiefe-maskentiefe+1)
```

in der Maskendefinition, jedoch ermöglicht sie die Verwendung derselben Maske für verschiedene Anzeigegrößen. JUSTIFY=BOTTOM wird ignoriert, wenn die Anzahl der Zeilen nicht ebenfalls angegeben wird. Wenn sowohl JUSTIFY=BOTTOM als auch LINE angegeben werden, wird der in LINE angegebene Wert ignoriert.

LEFT und RIGHT schließen sich gegenseitig aus, ebenso wie FIRST und LAST. Wenn FIRST oder LAST nicht angegeben werden, werden die Daten an der nächsten verfügbaren Position zugeordnet, wie dies durch andere Parameter der Maskendefinition und der aktuellen Zuordnungsoperation bestimmt wird. FIRST oder LAST wird ignoriert, sofern nicht ACCUM in Befehlen SEND MAP angegeben wird. Ansonsten wird nur eine Maske auf jeder Seite angeordnet.

Anmerkung: Wenn ein Feld durch eine Ausgabezuordnung (Ausgabemaske) aus einer anderen Quelle initialisiert wird, überschreiben Daten, die als Eingabe getippt werden, nur die entsprechende Länge der vorhandenen Daten. Darüber hinaus vorhandene Daten verbleiben in dem Feld und könnten zu unerwarteten Interpretationen der neuen Daten führen.

LINE

Gibt die Anfangszeile auf einer Seite an, in der Daten für eine Maske formatiert werden sollen.

NUMBER

Ist ein Wert im Bereich von 1-240, der eine Anfangszeilennummer angibt. Eine Anforderung zur Zuordnung von Daten in einer Zeile und einer Spalte, die als Antwort auf einen vorherigen BMS-Befehl formatiert wurden, bewirkt, dass die aktuelle Seite so behandelt wird, als wäre sie vollständig. Die neuen Daten werden in der angeforderten Zeile und Spalte auf der neuen Seite formatiert.

NEXT

Gibt an, dass die Formatierung von Daten auf der nächsten verfügbaren leeren Zeile beginnen soll. Wenn LINE=NEXT im Makro DFHMDI angegeben wird, wird diese Angabe für Eingabeoperationen ignoriert und LINE=1 angenommen.

SAME

Gibt an, dass die Formatierung von Daten auf derselben Zeile beginnen soll, die auch für einen vorhergehenden BMS-Befehl verwendet wird. Wenn COLUMN=NEXT angegeben wird, wird diese Angabe für Eingabeoperationen ignoriert und COLUMN=1 angenommen. Wenn die Daten nicht auf dieselbe Zeile passen, werden sie auf der nächsten verfügbaren Zeile platziert, die leer ist.

MAPATTS

Gibt die Attributtypen an, die in die physische Maske eingeschlossen werden sollen. Diese Typen können einer oder mehrere der folgenden sein: COLOR, HIGHLIGHT, OUTLINE, PS, SOSI, TRANSP und VALIDN. Diese Liste muss alle Attributtypen einschließen, die für einzelne Felder in der Maske (Makro DFHMDF) angegeben werden sollen.

Wenn möglich, werden diese Werte aus Operanden abgeleitet, die in den Makros DFHMDF und DFHMSD bereits angegeben sind. Wenn zum Beispiel COLOR=BLUE angegeben wurde, wird MAPATTS=COLOR angenommen.

OBFMT

Gibt an, ob externe Formatierung verwendet werden soll. Dieser Operand ist nur für logische 3650-Einheiten oder für einen Prozessor der Serie 8100, der mit DPS Release 2 ausgeführt wird und in CICS als logische Einheit vom LU-Typ 2 (LUTYPE2) definiert ist, verfügbar. Weitere Informationen finden Sie unter [Externe Formatierung](#).

Der Operand OBFMT überschreibt den Operanden OBFMT im Makro DFHMSD.

YES

Gibt an, dass diese Maskendefinition in der externen Formatierung verwendet werden kann.

NO

Gibt an, dass diese Maskendefinition in der externen Formatierung nicht verwendet werden kann.

OUTLINE

Ermöglicht, dass Linien über, unter, links oder rechts von einem Feld eingeschlossen werden. Mithilfe einer beliebigen Kombination dieser Linien können Sie Rahmen um Felder oder Gruppen von Feldern erstellen.

PARTN

Gibt die Standardpartition an, die Masken in dieser Maskengruppe zugeordnet werden soll. Wenn die Option ACTIVATE angegeben wird, wird die angegebene Partition auch dann aktiviert, wenn Masken in dieser Maskengruppe an ein Terminal ausgegeben werden, das Partitionen unterstützt.

Diese Option überschreibt die Option PARTN des Makros DFHMSD und wird durch jede Option OUTPARTN oder ACTPARTN im Befehl SEND MAP oder durch die Option INPARTN in einem Befehl RECEIVE MAP überschrieben.

Die Option PARTN wird ignoriert, wenn das Zielterminal Partitionen nicht unterstützt oder wenn der Transaktion keine Partitionsgruppe zugeordnet ist.

PS

Gibt an, dass programmierte Symbole verwendet werden sollen. Dies überschreibt den Operanden PS des Makros DFHMSD und wird durch den Operanden PS des Makros DFHMDF überschrieben.

BASE

Gibt an, dass der Basissymbolsatz verwendet werden soll.

psid

Gibt ein einzelnes EBCDIC-Zeichen oder einen Hexadezimalcode der Form X'nn' an, der den zu verwendenden Satz programmierter Symbole bezeichnet.

Der Operand PS wird ignoriert, wenn das Terminal keine programmierten Symbole unterstützt, wie dies durch die Option PROGSYMBOLS(YES) in der RDO-TYPETERM-Definition angegeben wird.

SIZE

Gibt die Größe einer Maske an.

zeile

Ist ein Wert aus dem Bereich von 1 bis 240, der die Tiefe einer Maske als Anzahl von Zeilen angibt.

spalte

Ist ein Wert aus dem Bereich von 1 bis 240, der die Breite einer Maske als Anzahl von Spalten angibt.

Dieser Operand ist in den folgenden Fällen erforderlich:

- Ein zugeordnetes Makro DFHMDF mit dem Operanden POS wird verwendet.
- Auf die Maske soll in einem Befehl SEND MAP mit der Option ACCUM verwiesen werden.
- Die Maske soll verwendet werden, wenn auf Eingabedaten von einem anderen Terminal als einem 3270-Terminal in einem Befehl RECEIVE MAP verwiesen wird.
- Die Maske soll verwendet werden, um Daten über CICS 3270 Web Bridge zu senden oder zu empfangen.

SOSI

Gibt an, dass das Feld eine Mischung aus EBCDIC- und DBCS-Daten enthalten kann. Die DBCS-Unterfelder in einem EBCDIC-Feld werden durch ein DBCS-Startzeichen (SO - Shift-out) und ein DBCS-Endezeichen (SI - Shift-in) begrenzt. Das Start- und das Endezeichen nehmen beide eine einzelne Anzeigeposition ein (die normalerweise als Leerzeichen angezeigt wird). Sie können in einer Ausgabe in ein beliebiges Nicht-DBCS-Feld eingeschlossen werden, sofern sie paarig korrekt angegeben werden. Der Terminalbenutzer kann sie eingehend übertragen, wenn sie bereits im Feld vorhanden sind. Er kann sich einem EBCDIC-Feld jedoch nur dann hinzufügen, wenn das Feld das Attribut SOSI besitzt.

TERM

Wurde aus Gründen der Kompatibilität mit früheren Releases beibehalten.

TIOAPFX

Gibt an, ob BMS einen Füller in die symbolischen Beschreibungsmasken einschließen soll, um das nicht verwendete TIOA-Präfix zuzulassen. Dieser Operand überschreibt den Operanden TIOAPFX, der für das Makro DFHMSD angegeben oder als Standardwert angenommen wurde. Wenn er nicht angegeben wird, wird der im Makro DFHMSD angegebene oder als Standard angenommene Wert verwendet.

YES

Gibt an, dass der Füller in die symbolische Beschreibungsmasken eingeschlossen werden soll. Verwenden Sie für Anwendungsprogramme auf Befehlsebene immer TIOAPFX=YES.

NO

Gibt an, dass der Füller nicht eingeschlossen werden soll.

TRAILER

Ermöglicht, dass die Maske bei der Seitenerstellung verwendet werden kann, ohne die Bedingung OVERFLOW zu beenden. Dieser Operand kann für mehr als eine Maske in einer Maskengruppe angegeben werden. Wenn eine Trailermaske in einer anderen als der Überlaufumgebung verwendet wird, wird der Platz, der normalerweise für Überlauftrailermasken reserviert wird, während der Zuordnung der Trailermaske nicht reserviert.

TRANSP

Bestimmt, ob der Hintergrund eines alphanumerischen Felds transparent oder nicht transparent (opak) ist, das heißt, ob ein darunter liegender (grafischer) Darstellungsbereich zwischen den Zeichen sichtbar ist.

VALIDN

Gibt Folgendes an:

- Dass eine Prüfung (Validierung) auf einem 8775-Terminal verwendet wird.
- Dass dieses Feld von globalen BMS-Benutzerexits verarbeitet werden kann.

Dies wird durch den Operanden VALIDN des Makros DFHMDF überschrieben und überschreibt den Operanden VALIDN des Makros DFHMSD.

MUSTFILL

Gibt an, dass das Feld vollständig mit Daten gefüllt werden muss. Ein Versuch, den Cursor von dem Feld wegzubewegen, bevor es gefüllt wurde, oder Daten aus einem unvollständigen Feld zu übertragen, löst die Bedingung INHIBIT INPUT aus.

MUSTENTER

Gibt an, dass Daten in das Feld eingegeben werden müssen, es jedoch nicht völlig ausfüllen müssen. Ein Versuch, den Cursor von einem leeren Feld wegzubewegen, löst die Bedingung INHIBIT INPUT aus.

TRIGGER

Gibt an, dass dieses Feld ein Triggerfeld (Auslöserfeld) ist. Triggerfelder werden in Unterstützung für Triggerfeld beschrieben.

USEREXIT

Gibt an, dass dieses Feld durch die globalen BMS-Benutzerexits XBMIN und XBMOUT verarbeitet werden soll, wenn dieses Feld in einem 3270-Datenstrom empfangen oder übertragen wird, sofern der entsprechende Exit aktiviert ist.

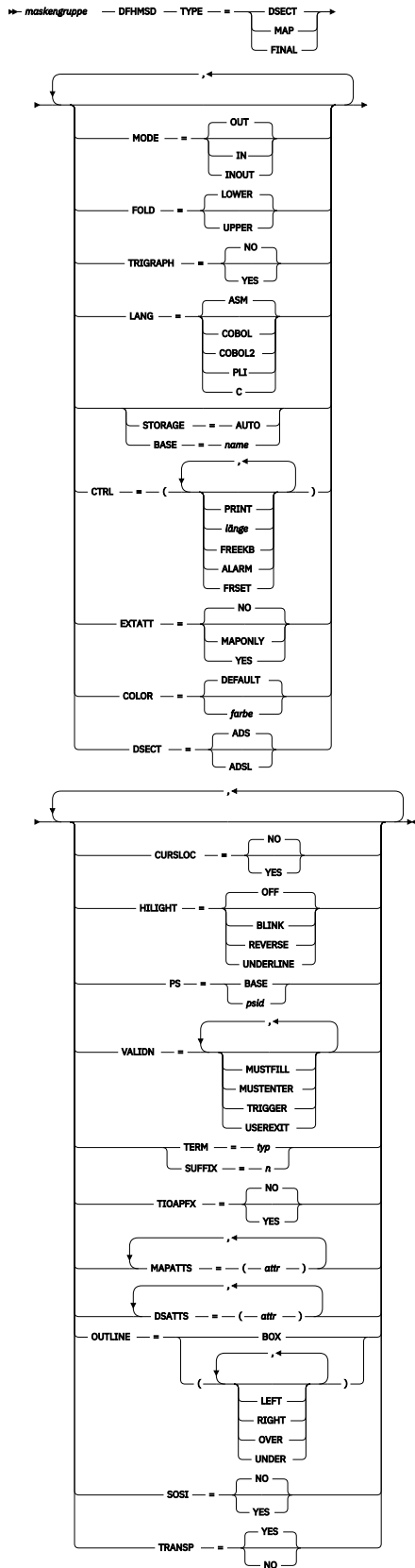
Die Operanden MUSTFILL, MUSTENTER und TRIGGER sind nur für Terminals gültig, die das erweiterte Attribut für Feldprüfung unterstützen. Andernfalls werden sie ignoriert. Die USEREXIT-Spezifikation gilt für alle 3270-Einheiten.

Anmerkung: Die USEREXIT-Spezifikation ist überhaupt nicht mit dem erweiterten Attribut für Feldprüfung verbunden, wie dies in der 3270-Datenstromarchitektur definiert ist.

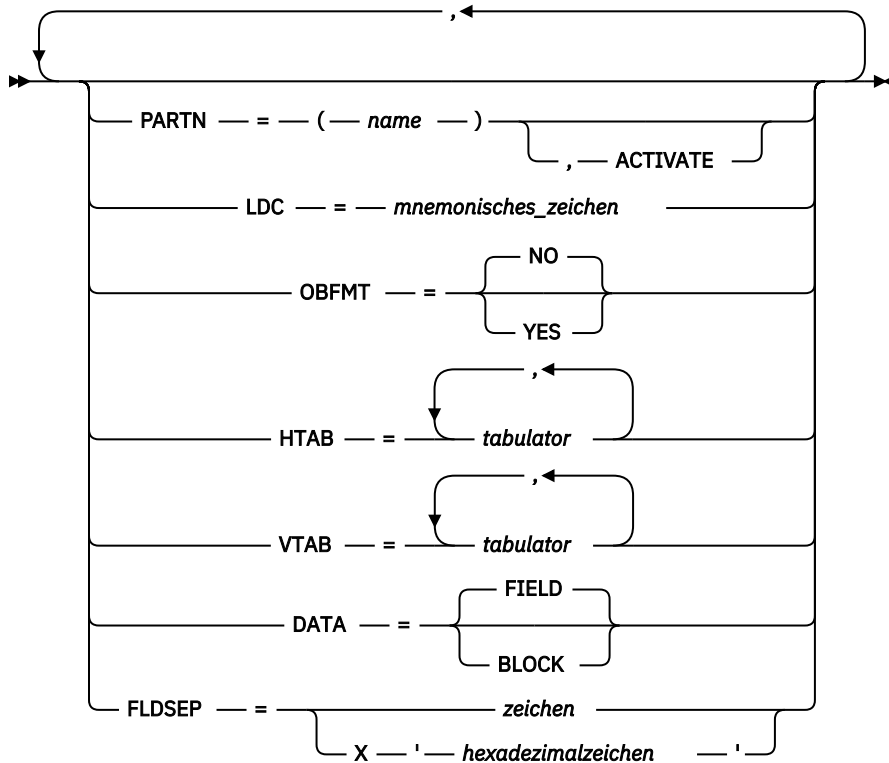
DFHMSD

Das Makro DFHMSD definiert eine Maskengruppe.

Minimaler BMS für DFHMSD



Standard-BMS für DFHMSD



Ein Makro DFHMSD definiert eine Maskengruppe (Mapset). Es beginnt mit dem folgenden Code:

```
DFHMSD TYPE=MAP      (oder TYPE=DSECT)
```

Es endet mit dem folgenden Code:

```
DFHMSD TYPE=FINAL
```

Der Wert *maskengruppe* ist der Name der Maskengruppe. Normalerweise ist der Name bis zu sieben Zeichen lang. Wenn die Maskengruppe jedoch zum Generieren von HTML-Vorlagen verwendet wird und mehr als 36 Masken (Maps) enthält, darf der Name die Länge von sechs Zeichen nicht überschreiten.

Ein Makro DFHMSD enthält ein oder mehrere Maskendefinitionsmakros, die wiederum jeweils ein oder mehrere Felddefinitionsmakros enthalten.

Stellen Sie sicher, dass die Namen von Masken und die Namen von Feldern in einer Maskengruppe (bzw. in mehreren Maskengruppen, die in dasselbe Anwendungsprogramm kopiert werden) eindeutig sind. Eine Maske kann jedoch den gleichen Namen wie eine Maskengruppe haben.

Bevor CICS eine physische Maske laden kann, müssen Sie eine physische Maske mit einem RDO-Befehl **DEFINE MAPSET** definieren.

Sie assemblieren eine BMS-Maskengruppendefinition, um entweder eine symbolische Beschreibungs-
maske oder eine physische Maske zu generieren. Informationen zum Assemblieren und Katalogisieren der Masken finden Sie unter .

Operanden

BASE

Gibt an, dass dieselbe Speicherbasis für die symbolischen Beschreibungsmasken aus mehr als einer Maskengruppe verwendet wird. Für jede Maskengruppe, die dieselbe Speicherbasis gemeinsam verwenden soll, wird derselbe Name angegeben. Da alle Maskengruppen mit derselben Basis denselben Speicher beschreiben, werden Daten, die zu einer zuvor verwendeten Maskengruppe gehören, möglicherweise überschrieben, wenn eine neue Maskengruppe verwendet wird. Verschiedene Masken in derselben Maskengruppe überlagern sich ebenfalls einander.

Dieser Operand ist für Assemblerprogramme nicht gültig und kann nicht verwendet werden, wenn STORAGE=AUTO angegeben wurde.

COLOR

Gibt die individuelle Farbe oder die Standardfarbe für die Maskengruppe an (soweit zutreffend). Dies wird durch den Operanden COLOR des Makros DFHMDI überschrieben, das wiederum durch den Operanden COLOR des Makros DFHMDF überschrieben wird.

Die gültigen Farben sind blau (blue), rot (red), rosa (pink), grün (green), türkis (turquoise), gelb (yellow) und neutral (neutral).

Der Operand COLOR wird ignoriert, wenn das Terminal keine Farbe unterstützt, wie dies durch die Option COLOR(YES) der RDO-TYPETERM-Definition angegeben wird.

CTRL

Definiert Merkmale von IBM 3270-Terminals. Die Verwendung **jeder** der Steueroptionen im Befehl SEND MAP überschreibt alle Steueroptionen im Makro DFHMDI, das wiederum alle Steueroptionen im Makro DFHMSD überschreibt.

Wenn CTRL mit der kumulativen BMS-Seitenverarbeitung (d. h. die Option ACCUM wird in BMS-Befehlen SEND MAP verwendet) verwendet wird, muss CTRL in der letzten (oder einzigen) Maske einer Seite angegeben werden, sofern dies nicht durch die Optionen ALARM, FREEKB usw. im Befehl SEND MAP oder im kumulierten Befehl SEND CONTROL überschrieben wird.

PRINT

Muss angegeben werden, wenn der Drucker gestartet werden soll. Wenn nicht angegeben, werden die Daten an den Druckerpuffer gesendet, jedoch nicht gedruckt. Dieser Operand wird ignoriert, wenn die Maskengruppe mit 3270-Anzeigen ohne Druckeradapterfunktion verwendet wird.

LENGTH

Gibt die Zeilenlänge auf dem Drucker an. Die Länge kann als L40, L64, L80 oder HONEOM angegeben werden. Die Angaben L40, L64 und L80 erzwingen eine neue Zeile nach 40, 64 bzw. 80 Zeichen. Bei Angabe von HONEOM wird die Standardzeilenlänge des Druckers verwendet. Wenn diese Option nicht angegeben wird, legt BMS die Zeilenlänge entsprechend der TCT-Seitengröße fest.

FREEKB

Veranlasst, dass die Tastatur entsperrt wird, nachdem die Maske geschrieben wurde. Wenn FREEKB nicht angegeben wird, bleibt die Tastatur gesperrt. Die Dateneingabe über die Tastatur wird unterbunden, bis dieser Status geändert wird.

ALARM

Aktiviert das akustische 3270-Signal. Für Nicht 3270-VTAM-Terminals setzt dieser Operand das Alarmflag im Funktionsverwaltungsholder (Function Management Header - FMH). (Diese Funktion wird von interaktiven Einheiten und logischen Einheiten mit Stapelverarbeitung nicht unterstützt).

FRSET

Gibt an, dass die Tags für modifizierte Daten (MDTs) aller Felder, die sich momentan im 3270-Puffer befinden, auf die Bedingung 'Nicht geändert' zurückgesetzt werden sollen (Feldzurücksetzung), bevor Maskendaten in den Puffer geschrieben werden. Dadurch kann das Makro DFHMDF mit dem Operanden ATTRB den finalen Status beliebiger Felder steuern, die als Antwort auf einen BMS-Befehl geschrieben oder erneut geschrieben werden.

CURSLOC

Gibt an, dass BMS für alle Operationen RECEIVE MAP mit dieser Maske auf 3270-Terminals ein Flag in dem Anwendungsdatenstrukturelement für das Feld setzt, in dem sich der Cursor befindet.

Das Flag kann von DFHBMCUR getestet werden (siehe Copybook DFHBMSCA in [Kapitel 8, „BMS-spezifische Konstanten“](#), auf Seite 841).

Das Flag kann wie folgt getestet werden (COBOL-Beispiel):

```
(DFHBMSCA)
...
02 DFHBMEOF PIC X VALUE X'80'.
02 DFHBMCUR PIC X VALUE X'02'.
02 DFHBMEC PIC X VALUE X'82'.
02 DFHBMFLG PIC X.
88 DFHERASE VALUES ARE X'80', X'82'.
88 DFHCURSR VALUES ARE X'02', X'82'.
MOVE FLD1F TO DFHBMFLG.
IF DFHERASE THEN ...
ELSE ...
IF DFHCURSR THEN ...
ELSE ...
```

Anmerkung:

1. Wenn CURSLOC=YES für die MAP-Definitionen angegeben wird und keine Daten für ein beliebiges Feld der Anwendungsdatenstruktur vorhanden sind, aber der Cursor in einem Feld liegt, das der Anwendungsdatenstruktur bekannt ist, setzt BMS das Cursor-Flag für das entsprechende Feld, jedoch sind die Daten für alle Felder in der Anwendungsdatenstruktur null und die Bedingung MAP-FAIL tritt nicht auf. Der in der Maske nicht zugeordnete Datenstrom ist für das Anwendungsprogramm nur verfügbar, wenn es sich um eine Anforderung RECEIVE DATA FROM handelt.
2. Eine gültige CURSLOC-Definition in DFHMDI überschreibt die Definition in DFHMSD.

DATA

Gibt das Format der Daten an.

FIELD

Gibt an, dass die Daten als fortlaufende Felder übergeben werden, wobei jedes Feld das folgende Format hat:

LL	A	Datenfeld
----	---	-----------

LL sind zwei Byte, die die Länge der Daten als Eingabe vom Terminal angeben (diese zwei Byte werden bei der Ausgabeverarbeitung ignoriert). A ist ein Byte, in dem der Programmierer ein Attribut platzieren kann, um das in der Maskengruppe angegebene Attribut zu überschreiben, das zur Verarbeitung dieser Daten verwendet wird (siehe Copybook DFHBMSCA in [Kapitel 8, „BMS-spezifische Konstanten“](#), auf Seite 841).

BLOCK

Gibt an, dass die Daten als fortlaufender Datenstrom im folgenden Format übergeben wird:

A	Datenfeld	Leerzeichen
---	-----------	-------------

Dieser Datenstrom wird als Zeilensegmente der Länge verarbeitet, die in der Maske angegeben wird, die zur Verarbeitung der Daten verwendet wird. Die Daten liegen in der Form vor, in der sie auf dem Terminal dargestellt werden. Das heißt, sie enthalten Datenfelder und dazwischen eingefügte Leerzeichen, die den Leerzeichen entsprechen, die zwischen den Feldern in der Ausgabe angezeigt werden sollen. Sie können DSATTS=YES nicht verwenden, wenn Sie DATA=BLOCK angeben.

Blockdaten werden unter eingehender erläutert.

DSATTS

Gibt die Attributtypen an, die in die symbolische Beschreibungsmaske eingeschlossen werden sollen. Diese Typen können einer oder mehrere der folgenden sein: COLOR, HILIGHT, OUTLINE, PS, SOSI,

TRANSP und VALIDN. Jeder Typ, der in DSATTS eingeschlossen wird, sollte auch in MAPATTS eingeschlossen werden.

DSECT

Gibt an, ob das generierte Copybook die Normalform oder die Langform der Anwendungsdatenstruktur (ADS) unterstützt.

ADS

Fordert die Zuordnung der Normalform der ADS an (Standardeinstellung).

ADSL

Fordert die Zuordnung der Langform der ADS an, in der alle Felder an 4-Byte-Grenzen ausgerichtet werden. Diese Form ist für die 3270-Brückenschnittstelle für IBM MQ erforderlich.

Diese Option erfordert LANG=C.

EXTATT

Dieser Operand wird aus Gründen der Kompatibilität mit früheren Releases unterstützt. Jedes erweiterte Attribute kann einzeln definiert werden. Verwenden Sie für neue Masken stattdessen die Operanden DSATTS und MAPATTS.

NO

Ist äquivalent dazu, weder den Operanden DSATTS noch den Operanden MAPATTS anzugeben.

YES

Ist äquivalent zur Angabe der folgenden Operanden:

```
MAPATTS=(COLOR,HILIGHT,PS,VALIDN)
DSATTS=(COLOR,HILIGHT,PS,VALIDN)
```

MAPONLY

Ist äquivalent zur Angabe des folgenden Operanden:

```
MAPATTS=(COLOR,HILIGHT,PS,VALIDN)
```

FLDSEP

Gibt die Feldtrennzeichensequenz (1–4 Zeichen) für Eingaben über Nicht-3270-Einheiten an. Eingaben über Nicht-3270-Einheiten können als einzelne Zeichenfolge von Daten mit der Feldtrennzeichensequenz als Feldbegrenzung eingegeben werden. Die Daten zwischen den Feldtrennzeichen werden in der Reihenfolge in die Eingabefelder der Maske versetzt.

FOLD

Gibt an, ob in C-Programmen Klein- oder Großbuchstaben generiert werden sollen.

FOLD ist nur für in C geschriebene Programme verfügbar.

HILIGHT

Gibt das Standardhervorhebungsattribut für alle Felder in allen Masken einer Maskengruppe an. Dies wird durch den Operanden HILIGHT des Makros DFHMDI überschrieben, das wiederum durch den Operanden HILIGHT des Makros DFHMDF überschrieben wird.

OFF

Gibt an, dass keine Hervorhebung verwendet wird (Standardeinstellung).

BLINK

Gibt an, dass das Feld blinken muss.

REVERSE

Gibt an, dass das Zeichen oder Feld in Umkehranzeige dargestellt wird. Beispiel: Auf einem 3278 schwarze Zeichen auf grünem Hintergrund.

UNDERLINE

Gibt an, dass ein Feld unterstrichen wird.

Der Operand HILIGHT wird ignoriert, wenn das Terminal keine Hervorhebung unterstützt, wie dies durch die Option HILIGHT(YES) in der RDO-TYPETERM-Definition angegeben wird.

HTAB

Gibt eine oder mehrere Tabulatorpositionen zur Verwendung mit interaktiven logischen Einheiten und logischen Einheiten mit Stapelverarbeitung sowie mit SCS-Druckern mit horizontaler Formularvor-schubsteuerung an.

LANG

Gibt die Quellsprache des Anwendungsprogramms an, in das die symbolischen Beschreibungsmas-ken in der Maskengruppe kopiert werden. COBOL ist OS/VS COBOL, das unter dieser CICS-Version nicht ausgeführt werden kann. COBOL2 ist entweder Enterprise COBOL oder VS COBOL II. Diese Opti-on muss nur für das Makro DFHMSD mit TYPE=DSECT codiert werden. Wenn eine Maskengruppe von mehr als einem Programm verwendet werden soll und die Programme nicht alle in derselben Quellen-sprache geschrieben sind, muss eine separate Version der Maskengruppe für jede Programmierspra-che definiert werden.

LDC

Gibt den Code an, der von CICS verwendet werden soll, um den mnemonischen Code für logische Ein-heiten zu bestimmen, der für eine BMS-Ausgabeoperation verwendet werden soll. Wenn in keiner vor-herigen BMS-Ausgabe in der logischen Nachricht ein Operand LDC angegeben wurde, wird dieser Operand LDC im Funktionsverwaltungsheader an die logische Einheit übertragen. Dieser Operand wird nur für von VTAM unterstützte 3600-Terminals und für logische Einheiten mit Stapelverarbeitung unterstützt.

MAPATTS

Gibt die Attributtypen an, die in die physische Maske eingeschlossen werden sollen. Diese Typen kön-nen einer oder mehrere der folgenden sein: COLOR, HILIGHT, OUTLINE, PS, SOSI, TRANSP und VA-LIDN. Diese Liste muss alle Attributtypen einschließen, die für einzelne Felder in der Maske (Makro DFHMDf) angegeben werden sollen.

Wenn möglich, werden diese Werte aus Operanden abgeleitet, die in den Makros DFHMDI und DFHMSD bereits angegeben sind. Wenn zum Beispiel COLOR=BLUE angegeben wurde, wird MA-PATTS=COLOR angenommen.

MODE

Gibt an, ob die Maskengruppe für Eingaben, für Ausgaben oder für beides verwendet werden soll.

OBFMT

Gibt an, ob externe Formatierung verwendet werden soll. Dieser Operand ist nur für logische 3650-Einheiten oder für einen Prozessor der Serie 8100, der mit DPS Release 2 ausgeführt wird und in CICS als logische Einheit vom LU-Typ 2 (LUTYPE2) definiert ist, verfügbar. Weitere Informationen finden Sie unter .

Der Operand OBFMT im Makro DFHMSD wird durch den Operanden OBFMT im Makro DFHMDI über-schrieben.

YES

Gibt an, dass alle Masken in dieser Maskengruppe in einer externen Formatierung verwendet wer-den können, mit Ausnahme derer, für OBFMT=NO im Makro DFHMDI angegeben wurde.

NO

Gibt an, dass keine Masken in dieser Maskengruppe in einer externen Formatierung verwendet werden können, mit Ausnahme derer, für OBFMT=YES im Makro DFHMDI angegeben wurde.

OUTLINE

Ermöglicht, dass Linien über, unter, links oder rechts von einem Feld eingeschlossen werden. Mithilfe einer beliebigen Kombination dieser Linien können Sie Rahmen um Felder oder Gruppen von Feldern erstellen.

PARTN

Gibt die Standardpartition an, die Masken in dieser Maskengruppe zugeordnet werden soll. Wenn die Option ACTIVATE angegeben wird, wird die angegebene Partition auch dann aktiviert, wenn Masken in dieser Maskengruppe an ein Terminal ausgegeben werden, das Partitionen unterstützt. Diese Option wird durch den Operanden PARTN des Makros DFHMDI überschrieben, der wiederum durch jede Op-tion OUTPARTN oder ACTPARTN im Befehl SEND MAP oder durch die Option INPARTN in einem Be-fehl RECEIVE MAP überschrieben wird.

Der Operand PARTN wird ignoriert, wenn das Zielterminal Partitionen nicht unterstützt oder wenn der Transaktion keine Partitionsgruppe zugeordnet ist.

PS

Gibt an, dass programmierte Symbole verwendet werden sollen. Dies wird durch den Operanden PS des Makros DFHMDI überschrieben, das wiederum durch den Operanden PS des Makros DFHMDF überschrieben wird.

BASE

Gibt an, dass der Basissymbolsatz verwendet werden soll.

psid

Gibt ein einzelnes EBCDIC-Zeichen oder einen Hexadezimalcode der Form X'nn' an, der den zu verwendenden Satz programmierter Symbole bezeichnet.

Der Operand PS wird ignoriert, wenn das Terminal keine programmierten Symbole unterstützt, wie dies durch PROGSYMBOLS(YES) in der RDO-TYPETERM-Definition angegeben wird.

SOSI

Gibt an, dass das Feld eine Mischung aus EBCDIC- und DBCS-Daten enthalten kann. Die DBCS-Unterfelder in einem EBCDIC-Feld werden durch ein DBCS-Startzeichen (SO - Shift-out) und ein DBCS-Endezeichen (SI - Shift-in) begrenzt. Das Start- und das Endezeichen nehmen beide eine einzelne Anzeigeposition ein (die normalerweise als Leerzeichen angezeigt wird). Sie können in einer Ausgabe in ein beliebiges Nicht-DBCS-Feld eingeschlossen werden, sofern sie paarig korrekt angegeben werden. Der Terminalbenutzer kann sie eingehend übertragen, wenn sie bereits im Feld vorhanden sind. Er kann sich einem EBCDIC-Feld jedoch nur dann hinzufügen, wenn das Feld das Attribut SOSI besitzt.

STORAGE

Die Bedeutung dieses Operanden hängt wie folgt von der Sprache ab, in der Anwendungsprogramme geschrieben sind:

Für ein Programm in der Sprache **COBOL** gibt STORAGE=AUTO an, dass die symbolische Beschreibungsmaske in der Maskengruppe separate (d. h. nicht neu definierte) Bereiche von Speicher belegen sollen. Dieser Operand wird verwendet, wenn die symbolischen Beschreibungsmasken in den Arbeitspeicherabschnitt kopiert werden und der Speicher für die separaten Masken in der Maskengruppe gleichzeitig verwendet werden soll.

Für ein Programm in der Sprache **C** gibt STORAGE=AUTO an, dass die symbolischen Beschreibungsmasken so zu definieren sind, dass sie die Klasse für automatischen Speicher haben. Wenn STORAGE=AUTO nicht angegeben wird, werden sie als Zeiger deklariert.

Für ein Programm in der Sprache **PL/I** gibt STORAGE=AUTO an, dass die symbolischen Beschreibungsmasken so zu deklarieren sind, dass sie die Speicherklasse AUTOMATIC haben. Wenn STORAGE=AUTO nicht angegeben wird, werden sie als BASED deklariert.

Für ein Programm in **Assemblersprache** gibt STORAGE=AUTO an, dass einzelne Masken in einer Maskengruppe separate Bereiche von Speicher belegen sollen, anstatt sich gegenseitig zu überlagern.

Für alle Sprachen:

- Sie können nicht BASE=name und STORAGE=AUTO für dieselbe Maskengruppe angeben.
- Wenn STORAGE=AUTO angegeben wird und TIOAPFX nicht angegeben wird, wird TIOAPFX=YES angenommen.

SUFFIX

Gibt ein aus einem Zeichen bestehendes, benutzerdefiniertes und einheitenabhängiges Suffix für diese Maskengruppe an, das als Alternative zu einem durch den Operanden TERM generierten Suffix verwendet wird. Das durch diesen Operanden angegebene Suffix muss dem Wert einer Transaktion, die im Attribut ALTSUFFIX einer TYPETERM-Definition definiert ist, oder dem Wert von ALTSFX in der Terminalsteuertabelle (TYPE=TERMINAL) entsprechen. Verwenden Sie einen numerischen Wert, um Konflikte mit Suffixen zu vermeiden, die durch den Operanden TERM generiert werden.

TERM

Gibt den Typ von Terminal oder logischer Einheit (LU) an, der der Maskengruppe zugeordnet ist. Wenn kein Terminaltyp und keine logische Einheit angegeben wird, wird 3270 angenommen. Die Terminal-

typen und logischen Einheiten, die Sie angeben können, werden zusammen mit ihren generierten Suffixen in Tabelle 26 auf Seite 879 aufgeführt.

Darüber hinaus sollten Sie Folgendes beachten:

Wenn ALL angegeben wird, stellen Sie sicher, dass geräteabhängige Zeichen nicht in die Maskengruppe eingeschlossen werden und dass Formatmerkmale wie Seitengröße für alle Ein-/Ausgabeoperationen (und alle Terminals) geeignet sind, in denen die Maskengruppe angewendet wird. Einige Terminals sind zum Beispiel auf 480 Byte, andere auf 1920 Byte beschränkt. Das Tastaturanzeigegerät 3604 ist auf sechs Zeilen von je 40 Zeichen begrenzt. Innerhalb dieser Leitlinien kann die Verwendung von ALL wichtige Vorteile bieten. Da eine Assemblierungsausführung für jede Maskengenerierung erforderlich ist, kann die Verwendung von ALL, die angibt, dass eine Maske für mehr als ein Terminal verwendet werden soll, zu erheblichen Zeit- und Speichereinsparungen führen.

Allerdings lässt sich eine bessere Laufzeitleistung für Masken, die von einzelnen Terminaltypen verwendet werden, erzielen, wenn der Terminaltyp (und nicht ALL) angegeben wird. Alternativ kann die BMS-Unterstützung für einheitenabhängige Maskengruppen durch die Angabe von NODDS im Operanden BMS der Systeminitialisierungsparameter umgangen werden.

<i>Tabelle 26. BMS-Terminaltypen</i>		
Typ	Suffix	Anmerkungen
CRLP	A	Kartenleser-Eingabe/Zeilendrucker-Ausgabe
TAPE	B	
DISK	C	
TWX	D	
1050	E	
2740	F	
2741	G	
2770	I	
2780	J	
3780	K	
3270-1 (40 Spalten)	L	
3270-2 (80 Spalten)	M	
INTLU/3767/3770I/SCS	p	Alle interaktiven logischen Einheiten (LUs), einschließlich LU 3790 mit vollem Funktionsumfang und SCS-Drucker-LUs (3270 und 3790).
2980	Q	
2980-4	R	
<u>3270</u>	leer	Standardwert, wenn TERM nicht angegeben wird. Äquivalent zu ALL; verwendet, wenn Modelle nicht unterschieden werden müssen.
3601	U	
3653	V	Und LU (3653) mit Hostdialog.
3650UP	W	Und Interpreter-LU.
3650/3270	X	Und LU (3270) mit Hostdialog.
BCHLU/3770B	Y	Und alle Stapelverarbeitungs- und BDI-LUs.
ALL (alle obigen)	leer	

TIOAPFX

Gibt an, ob BMS einen Füller in die symbolischen Beschreibungsmasken einschließen soll, um das nicht verwendete TIOA-Präfix zuzulassen.

YES

Gibt an, dass der Füller in die symbolische Beschreibungsmasken eingeschlossen werden soll. Wenn TIOAPFX=YES angegeben wird, haben alle Masken in der Maskengruppe den Füller, außer wenn TIOAPFX=NO im Makro DFHMDI angegeben wird. Die Angabe TIOAPFX=YES ist die Standardeinstellung, wenn STORAGE=AUTO angegeben wird. TIOAPFX=YES sollte **immer** für Anwendungsprogramme auf Befehlsebene verwendet werden.

NO

Ist der Standardwert, sofern nicht STORAGE=AUTO angegeben wird, und gibt an, dass der Füller nicht eingeschlossen werden soll. Der Füller kann für eine Maske trotzdem eingeschlossen werden, wenn TIOAPFX=YES im Makro DFHMDI angegeben wird.

TRANSP

Bestimmt, ob der Hintergrund eines alphanumerischen Felds transparent oder nicht transparent (opak) ist, das heißt, ob ein darunter liegender (grafischer) Darstellungsbereich zwischen den Zeichen sichtbar ist.

TRIGRAPH

Gibt aus drei Zeichen bestehende Zeichenfolgen (Trigrafen) an, die in symbolischen Masken in der Programmiersprache C verwendet werden sollen.

Bei TRIGRAPH=YES werden Trigraphfolgen generiert:

```
{
    Wird als ??< ausgegeben.
}
    Wird als ??> ausgegeben.
[
    Wird als ??( ausgegeben.
]
    Wird als ??) ausgegeben.
```

Diese Option ist nur für in der Programmiersprache C geschriebene Programme verfügbar.

TYPE

Gibt den Typ von Maske an, der mit der Definition generiert werden soll. Beide Typen von Maske müssen generiert werden, bevor die Maskengruppe von einem Anwendungsprogramm verwendet werden kann. Wenn ausgerichtete symbolische Beschreibungsmasken erforderlich sind, müssen Sie sicherstellen, dass Sie SYSPARM=ADSECT und SYSPARM=AMAP angeben, wenn Sie die symbolischen bzw. physischen Masken assemblieren.

DSECT

Gibt an, dass eine symbolische Beschreibungsmaske generiert werden soll. Symbolische Beschreibungsmasken müssen in das Quellenprogramm kopiert werden, bevor es umgesetzt und kompiliert werden.

MAP

Gibt an, dass eine physische Maske generiert werden soll. Physische Masken müssen assembliert oder kompiliert, in Bezug auf ihre Links bearbeitet und in der CICS-Programmbibliothek katalogisiert werden, bevor sie von einem Anwendungsprogramm verwendet werden können.

Wenn sowohl eine physische Maske (MAP) als auch eine symbolische Beschreibungsmaske (DSECT) im selben Job generiert werden sollen, kann die Option SYSPARM im Ausführungsschritt für den Assemblerjob verwendet werden.

VALIDN

Gibt Folgendes an:

- Dass eine Prüfung (Validierung) auf einem 8775-Terminal verwendet wird.
- Dass dieses Feld von globalen BMS-Benutzerexits verarbeitet werden kann.

Dies wird durch den Operanden VALIDN des Makros DFHMDI überschrieben, das wiederum durch den Operanden VALIDN des Makros DFHMDF überschrieben wird.

MUSTFILL

Gibt an, dass das Feld vollständig mit Daten gefüllt werden muss. Ein Versuch, den Cursor von dem Feld wegzubewegen, bevor es gefüllt wurde, oder Daten aus einem unvollständigen Feld zu übertragen, löst die Bedingung INHIBIT INPUT aus.

MUSTENTER

Gibt an, dass Daten in das Feld eingegeben werden müssen, es jedoch nicht völlig ausfüllen müssen. Ein Versuch, den Cursor von einem leeren Feld wegzubewegen, löst die Bedingung INHIBIT INPUT aus.

TRIGGER

Gibt an, dass dieses Feld ein Triggerfeld (Auslöserfeld) ist. Triggerfelder werden in beschrieben.

USEREXIT

Gibt an, dass dieses Feld durch die globalen BMS-Benutzerexits XBMIN und XBMOUT verarbeitet werden soll, wenn dieses Feld in einem 3270-Datenstrom empfangen oder übertragen wird, sofern der entsprechende Exit aktiviert ist.

Die Operanden MUSTFILL, MUSTENTER und TRIGGER sind nur für Terminals gültig, die das erweiterte Attribut für Feldprüfung unterstützen. Andernfalls werden sie ignoriert. Die USEREXIT-Spezifikation gilt für alle 3270-Einheiten.

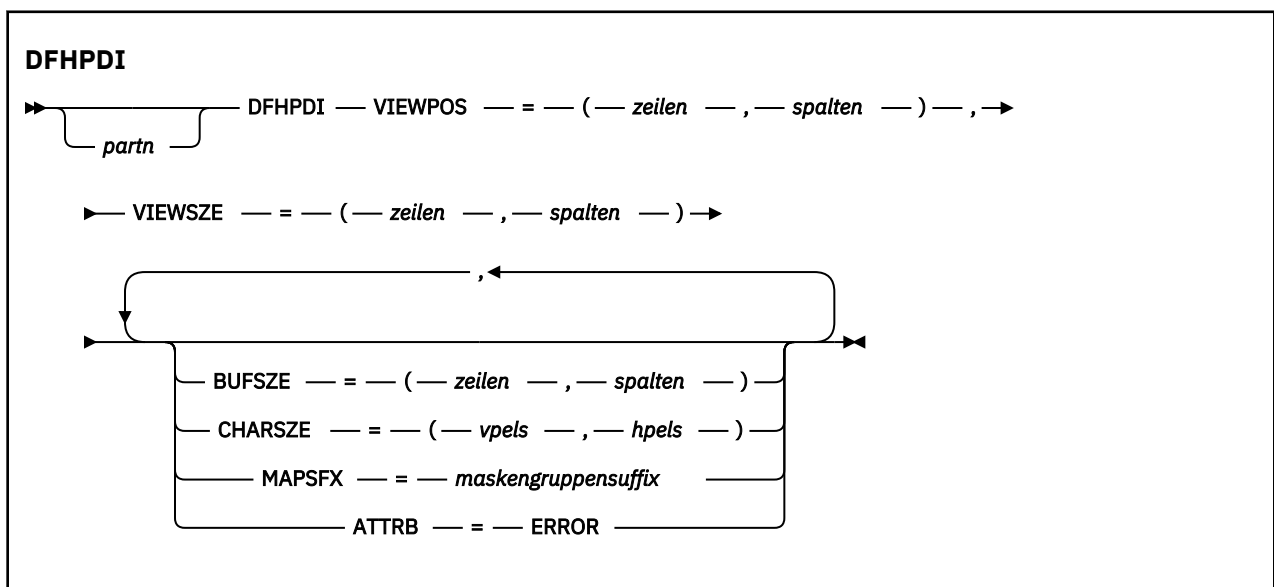
Anmerkung: Die USEREXIT-Spezifikation ist nicht mit dem erweiterten Attribut für Feldprüfung verbunden, wie dies in der 3270-Datenstromarchitektur definiert ist.

VTAB

Gibt eine oder mehrere Tabulatorpositionen zur Verwendung mit interaktiven logischen Einheiten und logischen Einheiten mit Stapelverarbeitung sowie mit SCS-Druckern mit vertikaler Formularvorschubsteuerung an.

DFHPDI

Eine Partitionsgruppe enthält eine oder mehr Partitionen. Jede Partition wird durch Codieren eines Partitionsdefinitions makros definiert.



Der Wert “partn” ist ein Partitionsname (1–2 Zeichen). Durch diesen Wert können Sie auf die Partition in Ihren Anwendungsprogrammen verweisen.

Jede Partition in einer Partitionsgruppe muss einen anderen Namen haben. Nur die Fehlerpartition kann unbenannt sein (siehe Operand ATTRB=ERROR).

Partitionen werden durch das Codieren der Makros DFHPSD (Partitionsgruppendifinition) und DFHPDI (Partitionsdefinition) definiert. Jede Partitionsdefinition muss Teil einer Partitionsgruppendifinition sein.

Operanden

ATTRB

Gibt an, dass Fehlernachrichten nach Möglichkeit an diese Partition übertragen werden. Der Inhalt der Partition wird gelöscht, bevor eine Fehlernachricht angezeigt wird. Die RDO-TYPETERM-Option ERRHIGHLIGHT wird berücksichtigt, jedoch wird die Option LASTLINE ignoriert.

BUFSZE(zeilen,spalten)

Gibt die Größe des Darstellungsbereichs für die Partition an. Einheitenbeschränkungen bedeuten, dass der Wert "spalten" gleich dem Wert "spalten" sein muss, der im Operanden VIEWSZE angegeben wird. Der Wert "zeilen" kann größer als oder, standardmäßig, gleich dem Wert sein, der im Operanden VIEWSZE angegeben wird. Ein größerer Wert für "zeilen" impliziert, dass das Zielterminal vertikales Blättern (Verschieben) unterstützt.

CHARSZE(vpels,hpels)

Gibt die Größe der Zeichenzelle an, die für jedes Zeichen, das in einer Partition angezeigt wird, reserviert werden soll. Sie geben die Größe als Anzahlen der vertikalen Bildelemente (vpels) und der horizontalen Bildelemente (hpels) an. Sie können diesen Operanden entweder nur im Makro DFHPSD oder sowohl im Makro DFHPSD als auch im Makro DFHPDI angeben. Die Werte, die im Makro DFHPSD angegeben werden, werden zu den Standardwerten für alle Partitionen in der Partitionsgruppe. Sie können diese Standardwerte für einzelne Partitionen überschreiben, indem Sie CHARSZE im Makro DFHPDI codieren.

MAPSFX(maskengruppensuffix)

Gibt das 1 Zeichen lange Maskengruppensuffix der Partition an. BMS verwendet das Suffix zur Auswahl von Maskengruppenversionen in derselben Weise wie für die RDO-Option ALTSUFFIX. Wenn dieser Operand weggelassen wird, wird das Suffix L angenommen, wenn der Wert "spalten" des Operanden BUFSZE kleiner oder gleich 40 ist. Andernfalls wird M angenommen.

VIEWPOS(zeilen,spalten)

Gibt die Position der linken oberen Ecke des Darstellungsfelds dieser Partition an. Sie geben die Position in Anzahl von Zeilen und Anzahl von Spalten an.

Das Makro DFHPDI prüft, ob sich die Darstellungsfelder überlappen. Wenn Sie die RDO-TYPETERM-Option ALTSCREEN oder den Operanden ALTSCRN des Makros DFHPSD codiert haben, prüft DFHPDI außerdem, ob alle Darstellungsfelder in den verwendbaren Bereich der Terminalanzeige passen.

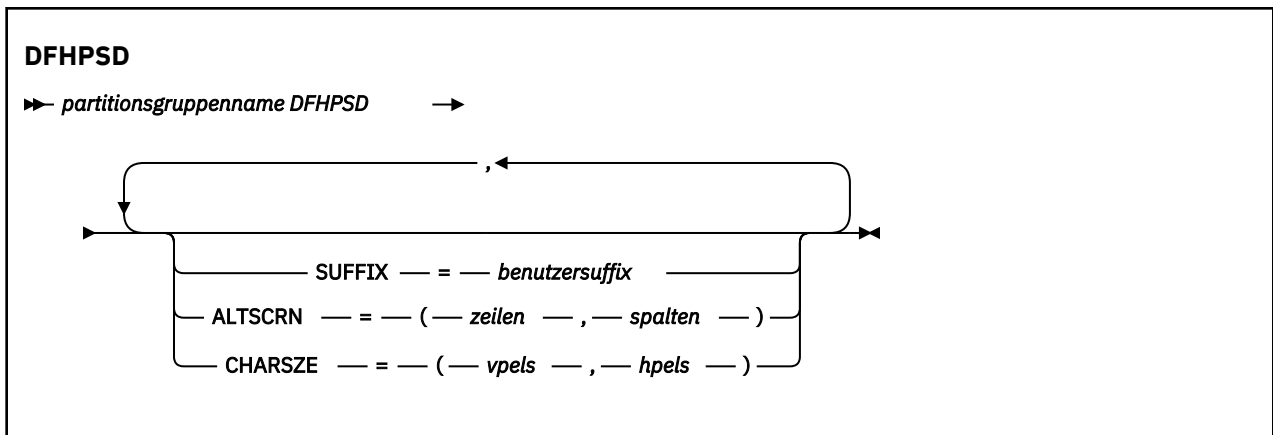
Anmerkung: Die hier gegebenen Informationen zur Positionierung von Darstellungsfeldern sind erforderlicherweise kurz. Weitere Informationen dazu finden Sie in der Komponentenbeschreibung für die Einheit, die Sie verwenden.

VIEWSZE(zeilen,spalten)

Gibt die Größe des Darstellungsbereichs der Partition in Zeilen und Spalten an. Das Makro DFHPDI prüft, ob sich die Darstellungsfelder überlappen. Wenn Sie die RDO-TYPETERM-Option ALTSCREEN oder den Operanden ALTSCRN des Partitionsgruppendifinitionsmakros DFHPSD codieren, prüft DFHPDI, ob die Partitionen alle in den verwendbaren Bereich der Anzeige passen.

DFHPSD

Jede Partitionsgruppendifinition enthält ein einzelnes Makro DFHPSD, auf das ein oder mehrere Makros DFHPDI sowie ein Partitionsgruppendifinitionsmakro DFHPSD TYPE=FINAL folgen.



Der Wert “partitionsgruppenname” ist ein Partitionsgruppenname (1 – 6 Zeichen).

Partitionen werden durch das Codieren der Makros DFHPSD (Partitionsgruppendifinition) und DFHPDI (Partitionsdefinition) definiert. Jede Partitionsdefinition muss Teil einer Partitionsgruppendifinition sein.

Operanden

ALTSCRN(zeilen,spalten)

Gibt die Größe des verwendbaren Bereichs des Zielterminals in Zeichen an. Dieser Wert stimmt normalerweise mit der Option ALTSCREEN der RDO-TYPETERM-Definition überein. Sie verwenden ALTSCRN, um sicherzustellen, dass die Darstellungsfelder von Partitionen in einer Partitionsgruppe in den verwendbaren Bereich der Anzeige passen.

CHARSE(vpels,hpels)

Gibt die Größe der Zeichenzelle an, die für jedes Zeichen, das in einer Partition angezeigt wird, reserviert werden soll. Sie geben die Größe als Anzahlen der vertikalen Bildelemente (vpels) und der horizontalen Bildelemente (hpels) an. Sie können diesen Operanden entweder nur im Makro DFHPSD oder sowohl im Makro DFHPSD als auch im Makro DFHPDI angeben. Die Werte, die in diesem Operanden angegeben werden, werden zu den Standardwerten für alle Partitionen in der Partitionsgruppe. Sie können diese Standardwerte für einzelne Partitionen überschreiben, indem Sie CHARSE im Makro DFHPDI codieren.

SUFFIX(benutzersuffix)

Gibt ein Benutzersuffix aus einem Zeichen für diese Version der Partitionsgruppe an. Durch das Suffix können verschiedenen Terminals verschiedene Versionen einer Partitionsgruppe zugeordnet werden. Wenn die Partitionsgruppe geladen werden soll, sucht CICS nach einer Version, deren Suffix der Angabe in der RDO-TYPETERM-Option ALTSUFFIX entspricht. Wenn die richtige Partitionsgruppenversion nicht gefunden wird, wird eine Version mit einem Standardsuffix (M oder L) geladen. Wenn auch keine Version mit Suffix gefunden wird, wird eine Version ohne Suffix geladen. Falls keine solche gefunden wird, wird die Operation mit APCT abgebrochen.

Abschließendes Makro DFHPSD

```
[partitionsgruppenname] DFHPSD TYPE=FINAL
```

Der Partitionsgruppenname (sofern angegeben) muss mit dem Namen übereinstimmen, der im Makro DFHPSD angegeben wird, das die Partitionsgruppendifinition gestartet hat.

Bemerkungen

Die vorliegenden Informationen wurden für Produkte und Services entwickelt, die auf dem deutschen Markt angeboten werden. IBM stellt dieses Material möglicherweise auch in anderen Sprachen zur Verfügung. Für den Zugriff auf das Material in einer anderen Sprache kann eine Kopie des Produkts oder der Produktversion in der jeweiligen Sprache erforderlich sein.

Möglicherweise bietet IBM die in dieser Dokumentation beschriebenen Produkte, Services oder Funktionen in anderen Ländern nicht an. Informationen über die gegenwärtig im jeweiligen Land verfügbaren Produkte und Services sind beim zuständigen IBM Ansprechpartner erhältlich. Hinweise auf IBM Lizenzprogramme oder andere IBM Produkte bedeuten nicht, dass nur Programme, Produkte oder Services von IBM verwendet werden können. Anstelle der IBM Produkte, Programme oder Services können auch andere, ihnen äquivalente Produkte, Programme oder Services verwendet werden, solange diese keine gewerblichen oder anderen Schutzrechte von IBM verletzen. Die Verantwortung für den Betrieb von Produkten, Programmen und Services anderer Anbieter liegt beim Kunden.

Für in diesem Handbuch beschriebene Erzeugnisse und Verfahren kann es IBM Patente oder Patentanmeldungen geben. Mit der Auslieferung dieses Handbuchs ist keine Lizenzierung dieser Patente verbunden. Lizenzanforderungen sind schriftlich an folgende Adresse zu richten (Anfragen an diese Adresse müssen auf Englisch formuliert werden):

*IBM Director of Licensing
IBM Europe, Middle East & Africa
Tour Descartes
2, avenue Gambetta
92066 Paris La Defense
France*

Trotz sorgfältiger Bearbeitung können technische Ungenauigkeiten oder Druckfehler in dieser Veröffentlichung nicht ausgeschlossen werden. Die hier enthaltenen Informationen werden in regelmäßigen Zeitabständen aktualisiert und als Neuausgabe veröffentlicht. IBM kann ohne weitere Mitteilung jederzeit Verbesserungen und/oder Änderungen an den in dieser Veröffentlichung beschriebenen Produkten und/oder Programmen vornehmen.

Verweise in diesen Informationen auf Websites anderer Anbieter werden lediglich als Service für den Kunden bereitgestellt und stellen keinerlei Billigung des Inhalts dieser Websites dar. Das über diese Websites verfügbare Material ist nicht Bestandteil des Materials für dieses IBM Produkt. Die Verwendung dieser Websites geschieht auf eigene Verantwortung.

Werden an IBM Informationen eingesandt, können diese beliebig verwendet werden, ohne dass eine Verpflichtung gegenüber dem Einsender entsteht.

Lizenznehmer des Programms, die Informationen zu diesem Produkt wünschen mit der Zielsetzung: (i) den Austausch von Informationen zwischen unabhängig voneinander erstellten Programmen und anderen Programmen (einschließlich des vorliegenden Programms) sowie (ii) die gemeinsame Nutzung der ausgetauschten Informationen zu ermöglichen, wenden sich an folgende Adresse:

*IBM Director of Licensing
IBM Corporation
North Castle Drive, MD-NC119
Armonk, NY 10504-1785
United States of America*

Die Bereitstellung dieser Informationen kann unter Umständen von bestimmten Bedingungen - in einigen Fällen auch von der Zahlung einer Gebühr - abhängig sein.

Die Lieferung des in diesen Informationen beschriebenen Lizenzprogramms sowie des zugehörigen Lizenzmaterials erfolgt auf der Basis der IBM Rahmenvereinbarung bzw. der Allgemeinen Geschäftsbedin-

gungen von IBM, der IBM Internationalen Nutzungsbedingungen für Programmpakete oder einer äquivalenten Vereinbarung.

Alle Informationen zu Produkten anderer Anbieter stammen von den Anbietern der aufgeführten Produkte, deren veröffentlichten Ankündigungen oder anderen allgemein verfügbaren Quellen. IBM hat diese Produkte nicht getestet und kann daher keine Aussagen zu Leistung, Kompatibilität oder anderen Merkmalen machen. Fragen zu den Leistungsmerkmalen von Produkten anderer Anbieter sind an den jeweiligen Anbieter zu richten.

Diese Veröffentlichung enthält Beispiele für Daten und Berichte des alltäglichen Geschäftsablaufs. Sie sollen nur die Funktionen des Lizenzprogramms illustrieren und können Namen von Personen, Firmen, Marken oder Produkten enthalten. Alle diese Namen sind frei erfunden und jede Ähnlichkeit mit Namen und Adressen tatsächlicher Personen oder Unternehmen ist rein zufällig.

COPYRIGHTLIZENZ:

Diese Veröffentlichung enthält Beispielanwendungsprogramme, die in Quellsprache geschrieben sind und Programmiertechniken in verschiedenen Betriebsumgebungen veranschaulichen. Sie dürfen diese Beispielprogramme kostenlos kopieren, ändern und verteilen, wenn dies zu dem Zweck geschieht, Anwendungsprogramme zu entwickeln, zu verwenden, zu vermarkten oder zu verteilen, die mit der Anwendungsprogrammierschnittstelle für die Betriebsumgebung konform sind, für die diese Beispielprogramme geschrieben werden. Diese Beispiele wurden nicht unter allen denkbaren Bedingungen getestet. Daher kann IBM die Zuverlässigkeit, Wartungsfreundlichkeit oder Funktion dieser Programme weder zusagen noch gewährleisten. Die Beispielprogramme werden ohne Wartung (auf "as-is"-Basis) und ohne jegliche Gewährleistung zur Verfügung gestellt. IBM übernimmt keine Haftung für Schäden, die durch die Verwendung der Beispielprogramme entstehen.

Informationen zu Programmierschnittstellen

Die von CICS zur Verfügung gestellte Dokumentation kann teilweise als Programmierschnittstelle betrachtet werden und zum Teil nicht.

Programmierschnittstellen, die es dem Kunden ermöglichen, Programme zur Nutzung der Services von CICS Transaction Server for z/OS, Version 5 Release 6 zu schreiben, sind in folgenden Abschnitten der Online-Produktdokumentation enthalten:

- [Anwendungen entwickeln](#)
- [Systemprogramme entwickeln](#)
- [CICS TS-Sicherheit](#)
- [Entwicklung für externe Schnittstellen](#)
- [Referenz zur Anwendungsentwicklung](#)
- [Referenz: Systemprogrammierung](#)
- [Referenz: Konnektivität](#)

Informationen, die NICHT zur Verwendung als Programmierschnittstelle von CICS Transaction Server for z/OS, Version 5 Release 6 bestimmt sind, die aber als Programmierschnittstelle missverstanden werden können, sind in folgenden Abschnitten der Online-Produktdokumentation enthalten:

- [Fehlerbehebung und Unterstützung](#)
- [CICS TS - Referenz zur Diagnose](#)

Wenn Sie auf die CICS-Dokumentation in Handbüchern im PDF-Format zugreifen, sind Programmierschnittstellen, die es dem Kunden ermöglichen, Programme zur Nutzung der Services von CICS Transaction Server for z/OS, Version 5 Release 6 zu schreiben, in den folgenden Handbüchern enthalten:

- Application Programming Guide und Application Programming Reference
- Business Transaction Services
- Customization Guide
- C++ OO Class Libraries

- Debugging Tools Interfaces Reference
- Distributed Transaction Programming Guide
- External Interfaces Guide
- Front End Programming Interface Guide
- IMS Database Control Guide
- Installation Guide
- Security Guide
- Supplied Transactions
- CICSplex SM Managing Workloads
- CICSplex SM Managing Resource Usage
- CICSplex SM Application Programming Guide and Application Programming Reference
- Java Applications in CICS

Wenn Sie auf die CICS-Dokumentation in Handbüchern im PDF-Format zugreifen, sind Informationen, die NICHT zur Verwendung als Programmierschnittstelle von CICS Transaction Server for z/OS, Version 5 Release 6 bestimmt sind, die aber als Programmierschnittstelle missverstanden werden können, in den folgenden Handbüchern enthalten:

- Data Areas
- Diagnosis Reference
- Problem Determination Guide
- CICSplex SM Problem Determination Guide

Marken

IBM, das IBM Logo und ibm.com sind Marken oder eingetragene Marken der International Business Machines Corporation in den USA und/oder anderen Ländern. Weitere Produkt- und Servicenamen können Marken von IBM oder anderen Unternehmen sein. Eine aktuelle Liste der IBM Marken finden Sie auf der Webseite [Copyright and trademark information](http://www.ibm.com/legal/copytrade.shtml) unter www.ibm.com/legal/copytrade.shtml.

Adobe, das Adobe-Logo, PostScript und das PostScript-Logo sind Marken oder eingetragene Marken von Adobe Systems Incorporated in den USA und/oder anderen Ländern.

Apache, Apache Axis2, Apache Maven, Apache Ivy, das Apache Software Foundation-Logo (ASF-Logo) und das Logo mit der ASF-Feder sind Marken der Apache Software Foundation.

Gradle und das Gradlephant-Logo sind eingetragene Marken von Gradle, Inc. oder der zugehörigen Tochtergesellschaften in den USA und/oder anderen Ländern.

Intel, das Intel-Logo, Intel Inside, das Intel Inside-Logo, Intel Centrino, das Intel Centrino-Logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium und Pentium sind Marken oder eingetragene Marken der Intel Corporation oder ihrer Tochtergesellschaften in den USA oder anderen Ländern.

Java und alle auf Java basierenden Marken und Logos sind Marken oder eingetragene Marken der Oracle Corporation und/oder ihrer verbundenen Unternehmen.

Die eingetragene Marke Linux[®] wird gemäß einer Unterlizenz der Linux Foundation verwendet, dem exklusiven Lizenznehmer von Linus Torvalds, dem Eigentümer der Marke auf einer weltweiten Basis.

Microsoft, Windows, Windows NT und das Windows-Logo sind Marken der Microsoft Corporation in den USA und/oder anderen Ländern.

Red Hat[®] und Hibernate[®] sind Marken oder eingetragene Marken von Red Hat, Inc. oder der zugehörigen Tochtergesellschaften in den USA und/oder anderen Ländern.

Spring Boot ist eine Marke von Pivotal Software, Inc. in den USA und/oder anderen Ländern.

UNIX ist eine eingetragene Marke von The Open Group in den USA und anderen Ländern.

Zowe[™], das Zowe-Logo und Open Mainframe Project[™] sind Marken von The Linux Foundation.

Nutzungsbedingungen für die Produktdokumentation

Die Berechtigungen zur Nutzung dieser Veröffentlichungen werden Ihnen auf der Basis der folgenden Bedingungen gewährt.

Anwendbarkeit

Diese Bedingungen sind eine Ergänzung der Nutzungsbedingungen auf der IBM Website.

Persönliche Nutzung

Sie dürfen diese Veröffentlichungen für Ihre persönliche, nicht kommerzielle Nutzung unter der Voraussetzung vervielfältigen, dass alle Eigentumsvermerke erhalten bleiben. Sie dürfen diese Veröffentlichungen oder Teile der Veröffentlichungen ohne ausdrückliche Genehmigung von IBM nicht weitergeben, anzeigen oder abgeleitete Werke davon erstellen.

Kommerzielle Nutzung

Sie dürfen diese Veröffentlichungen nur innerhalb Ihres Unternehmens und unter der Voraussetzung, dass alle Eigentumsvermerke erhalten bleiben, vervielfältigen, weitergeben und anzeigen. Sie dürfen diese Veröffentlichungen oder Teile der Veröffentlichungen ohne ausdrückliche Genehmigung von IBM außerhalb Ihres Unternehmens weder vervielfältigen, weitergeben oder anzeigen noch abgeleitete Werke davon erstellen.

Rechte

Abgesehen von den hier gewährten Berechtigungen werden keine weiteren Berechtigungen, Lizenzen oder Rechte (veröffentlicht oder stillschweigend) in Bezug auf die Veröffentlichungen oder darin enthaltene Informationen, Daten, Software oder geistiges Eigentum gewährt.

IBM behält sich das Recht vor, die hierin gewährten Berechtigungen nach eigenem Ermessen zurückzuziehen, wenn sich die Nutzung der Veröffentlichungen für IBM als nachteilig erweist oder wenn die obigen Nutzungsbestimmungen nicht genau befolgt werden.

Sie dürfen diese Informationen nur in Übereinstimmung mit allen anwendbaren Gesetzen und Vorschriften, einschließlich aller US-amerikanischen Exportgesetze und Verordnungen, herunterladen und exportieren.

IBM übernimmt keine Gewährleistung für den Inhalt dieser Veröffentlichungen. Diese Veröffentlichungen werden auf der Grundlage des gegenwärtigen Zustands (auf "as-is"-Basis) und ohne eine ausdrückliche oder stillschweigende Gewährleistung für die Handelsüblichkeit, die Verwendungsfähigkeit für einen bestimmten Zweck oder die Freiheit von Rechten Dritter zur Verfügung gestellt.

IBM Online-Datenschutzerklärung

IBM Softwareprodukte, einschließlich Software as a Service-Lösungen (*Softwareangebote*), können Cookies oder andere Technologien verwenden, um Informationen zur Produktnutzung zu erfassen, die Endbenutzererfahrung zu verbessern und Interaktionen mit dem Endbenutzer anzupassen oder zu anderen Zwecken. In vielen Fällen werden von den Softwareangeboten keine personenbezogenen Daten erfasst. Einige der IBM Softwareangebote können Sie jedoch bei der Erfassung personenbezogener Daten unterstützen. Wenn dieses Softwareangebot Cookies zur Erfassung personenbezogener Daten verwendet, sind nachfolgend nähere Informationen über die Verwendung von Cookies durch dieses Angebot zu finden:

Für die Webbenutzerschnittstelle von CICSplex System Manager (Hauptschnittstelle):

Abhängig von den bereitgestellten Konfigurationen kann dieses Softwareangebot Sitzungscookies und persistente Cookies zum Erfassen der Benutzernamen und anderer personenbezogener Daten einzelner Benutzer für das Sitzungsmanagement, die Authentifizierung, einen besseren Bedienungskomfort, zur Nutzungsüberwachung und für andere funktionale Zwecke verwenden. Diese Cookies können nicht inaktiviert werden.

Für die Webbenutzerschnittstelle von CICSplex System Manager (Datenschnittstelle):

Abhängig von den bereitgestellten Konfigurationen kann dieses Softwareangebot Sitzungscookies und persistente Cookies zum Erfassen der Benutzernamen und anderer personenbezogener Daten einzelner Benutzer für das Sitzungsmanagement, die Authentifizierung, einen besseren Bedienungskomfort, zur Nutzungsüberwachung und für andere funktionale Zwecke verwenden. Diese Cookies können nicht inaktiviert werden.

Für die Webbenutzerschnittstelle von CICSplex System Manager ("hello world"-Seite):

Abhängig von den bereitgestellten Konfigurationen kann dieses Softwareangebot Sitzungscookies verwenden, die keine personenbezogenen Daten erfassen. Diese Cookies können nicht inaktiviert werden.

Für CICS Explorer:

Abhängig von den bereitgestellten Konfigurationen kann dieses Softwareangebot persistente Vorgaben und Sitzungsvorgaben zum Erfassen der Benutzernamen und Kennwörter von Benutzern für das Sitzungsmanagement, die Authentifizierung und zur Single Sign-on-Konfiguration (einmalige Anmeldung) verwenden. Diese Vorgaben können nicht inaktiviert werden, auch wenn die Speicherung eines Benutzerkennworts auf einem Datenträger in verschlüsselter Form nur aktiviert werden kann, indem der Benutzer bei der Anmeldung explizit ein Kontrollkästchen aktiviert.

Wenn es die für dieses Softwareangebot bereitgestellten Konfigurationen Ihnen als Kunde ermöglichen, personenbezogene Daten von Endbenutzern über Cookies und andere Technologien zu erfassen, müssen Sie sich zu allen gesetzlichen Bestimmungen in Bezug auf eine solche Datenerfassung, einschließlich aller Mitteilungspflichten und Zustimmungsanforderungen, rechtlich beraten lassen.

Weitere Informationen zur Nutzung verschiedener Technologien, einschließlich Cookies, für diese Zwecke finden Sie unter [IBM Datenschutzrichtlinie](#) und in der [IBM Online-Datenschutzerklärung](#) im Abschnitt *Cookies, Web-Beacons und sonstige Technologien* sowie auf der Seite [IBM Softwareprodukte und Software-as-a-Service-Datenschutzerklärung](#).

Index

Sonderzeichen

- > 32K große COMMAREAs (Kanäle)
 - ASSIGN, Befehl [47](#)
 - DELETE CONTAINER (CHANNEL), Befehl [133](#)
 - MOVE CONTAINER (CHANNEL), Befehl [345](#)
 - PUT CONTAINER (CHANNEL), Befehl [355](#)

Numerische Stichwörter

- 2260, Anzeigestation
 - CONVERSE, Befehl [99](#)
 - RECEIVE, Befehl [425](#)
 - SEND, Befehl [497](#)
 - 2265, Anzeigestation
 - CONVERSE, Befehl [99](#)
 - RECEIVE, Befehl [425](#)
 - SEND, Befehl [497](#)
 - 2980, Terminalsystem für allgemeine Bankgeschäfte
 - Ausgabe in allgemeinen Puffer [427](#)
 - Ausgabesteuerung [427](#)
 - Befehle RECEIVE/SEND [426](#), [497](#)
 - DFH2980-Struktur [427](#)
 - Sparbuchsteuerung [426](#)
 - 32 K, COMMAREA-Größe über (Kanäle)
 - DELETE CHANNEL, Befehl [130](#)
 - GET CONTAINER (CHANNEL), Befehl [236](#)
 - Option CHANNEL des Befehls RETURN [460](#)
 - Option CHANNEL des Befehls XCTL [759](#)
 - QUERY CHANNEL, Befehl [363](#)
 - START CHANNEL, Befehl [579](#)
 - 3270, Informationsanzeigesystem
 - logische Einheit [87](#), [293](#), [417](#), [485](#)
 - 3600, Bankenkommunikationssystem
 - 3601, logische Einheit [87](#), [88](#), [418](#), [486](#)
 - 3614, logische Einheit [88](#), [418](#), [487](#)
 - logische 3601-Einheit [487](#)
 - Pipeline, logische Einheiten mit [486](#)
 - 3630, Werkskommunikationssystem (Plant Communication System)
 - Befehl SEND [487](#)
 - RECEIVE, Befehl [418](#)
 - 3650, Handelssystem
 - Interpreter, logische Einheit [299](#), [304](#)
 - 3650, Hostbefehlsprozessor
 - CONVERSE, Befehl [90](#)
 - 3650, Speichersystem
 - Hostdialog
 - logische 3270-Einheit [488](#)
 - logische 3653-Einheit [489](#)
 - logische Einheit für Interpreter [89](#), [419](#), [488](#)
 - 3650/3680, Speichersystem
 - Hostbefehlsprozessor, logische Einheit [489](#)
 - 3680, Hostbefehlsprozessor
 - Befehl CONVERSE [90](#)
 - 3680, programmierbares Speichersystem
 - Hostbefehlsprozessor, logische Einheit [489](#)
 - 3740, Dateneingabesystem [298](#), [299](#)
 - 3767, Kommunikationsterminal
 - interaktive logische Einheit [91](#), [420](#), [490](#)
 - 3770, Datenübertragungssystem
 - logische Einheit mit Stapelverarbeitung [91](#), [420](#)
 - 3790, Kommunikationssystem
 - 3270-Anzeige, logische Einheit [92](#), [491](#)
 - logische Einheit mit 3270-Anzeige [428](#)
 - logische Einheit mit SCS-Drucker [491](#)
 - logische Einheit mit vollem Funktionsumfang [421](#), [491](#)
 - voller Funktionsumfang, logische Einheit [92](#)
 - 64-Bit-Hauptspeicher initialisieren [248](#)
- ## A
- ABCODE, Option
 - ABEND, Befehl [27](#)
 - ASSIGN, Befehl [44](#)
 - CHECK ACQPROCESS, Befehl [76](#)
 - CHECK ACTIVITY, Befehl [78](#)
 - FETCH CHILD, Befehl [194](#), [195](#)
 - INQUIRE ACTIVITYID, Befehl [270](#)
 - ABDUMP, Option
 - ASSIGN, Befehl [44](#)
 - ABEND-Exit reaktivieren [263](#)
 - ABEND, Befehl [27](#)
 - Abfragen einer Datei (Dataset) [311](#)
 - Ablaufzeit, Benachrichtigung bei Erreichen [348](#)
 - Abnormale Beendigung, Befehle zur Unterstützung [17](#)
 - Abnormale Beendigung, Task [263](#)
 - ABOFFSET, Option
 - ASSIGN, Befehl [44](#)
 - ABPROGRAM, Option
 - ASSIGN, Befehl [44](#)
 - CHECK ACQPROCESS, Befehl [76](#)
 - CHECK ACTIVITY, Befehl [78](#)
 - INQUIRE ACTIVITYID, Befehl [270](#)
 - Abrufen von 64-Bit-Hauptspeicher [248](#)
 - Abrufen von gespeicherten Daten für eine Task [455](#)
 - Abrufen von Hauptspeicher [245](#)
 - Absolute Ausdrücke [7](#)
 - ABSTIME, Option
 - ASKTIME, Befehl [41](#)
 - CONVERTTIME, Befehl [104](#)
 - FORMATTIME, Befehl [199](#)
 - INQUIRE TIMER, Befehl [277](#)
 - ACCUM, Option
 - SEND CONTROL, Befehl [501](#)
 - SEND MAP, Befehl [507](#)
 - SEND TEXT, Befehl [521](#)
 - ACEE, Option
 - ADDRESS, Befehl [33](#)
 - ACQACTIVITY, Option
 - CANCEL, Befehl (BTS) [69](#)
 - CHECK ACTIVITY, Befehl [78](#)
 - DELETE CONTAINER (BTS), Befehl [132](#)
 - FORCE TIMER, Befehl [197](#)

ACQACTIVITY, Option (*Forts.*)
 GET CONTAINER (BTS), Befehl [234](#)
 LINK ACTIVITY, Befehl [335](#)
 PUT CONTAINER (BTS), Befehl [354](#)
 RESUME, Befehl [454](#)
 RUN, Befehl [478](#)
 SUSPEND (BTS), Befehl [598](#)

ACQPROCESS, Option
 CANCEL, Befehl (BTS) [69](#)
 CHECK ACQPROCESS, Befehl [76](#)
 DELETE CONTAINER (BTS), Befehl [132](#)
 FORCE TIMER, Befehl [197](#)
 GET CONTAINER (BTS), Befehl [234](#)
 LINK ACQPROCESS, Befehl [332](#)
 PUT CONTAINER (BTS), Befehl [354](#)
 RESET ACQPROCESS, Befehl [447](#)
 RESUME, Befehl [454](#)
 RUN, Befehl [478](#)
 SUSPEND (BTS), Befehl [598](#)

ACQUIRE, Befehl [28](#)

ACTION, Option
 WEB CONVERSE, Befehl [647](#)
 WEB SEND, Befehl (Client) [711](#)
 WEB SEND, Befehl (Server) [701](#)
 WRITE OPERATOR, Befehl [739](#)

ACTIVE, Modus einer Aktivität [271](#)

ACTIVITY, Option
 ASSIGN, Befehl [44](#)
 CANCEL, Befehl (BTS) [69](#)
 CHECK ACTIVITY, Befehl [78](#)
 DEFINE ACTIVITY, Befehl [105](#)
 DELETE ACTIVITY, Befehl [129](#)
 DELETE CONTAINER (BTS), Befehl [132](#)
 GET CONTAINER (BTS), Befehl [234](#)
 GETNEXT ACTIVITY, Befehl [252](#)
 INQUIRE ACTIVITYID, Befehl [270](#)
 LINK ACTIVITY, Befehl [335](#)
 PUT CONTAINER (BTS), Befehl [354](#)
 RESET ACTIVITY, Befehl [448](#)
 RESUME, Befehl [454](#)
 RUN, Befehl [478](#)
 SUSPEND (BTS), Befehl [598](#)

ACTIVITYBUSY, Bedingung
 ACQUIRE, Befehl [29](#)
 CANCEL, Befehl (BTS) [70](#)
 CHECK ACTIVITY, Befehl [79](#)
 DELETE ACTIVITY, Befehl [129](#)
 LINK ACTIVITY, Befehl [335](#)
 RESET ACTIVITY, Befehl [448](#)
 RESUME, Befehl [454](#)
 RUN, Befehl [479](#)
 SUSPEND (BTS), Befehl [598](#)

ACTIVITYERR, Bedingung
 ACQUIRE, Befehl [29](#)
 CANCEL, Befehl (BTS) [70](#)
 CHECK ACTIVITY, Befehl [80](#)
 DEFINE ACTIVITY, Befehl [106](#)
 DELETE ACTIVITY, Befehl [129](#)
 DELETE CONTAINER (BTS), Befehl [132](#)
 GET CONTAINER (BTS), Befehl [235](#)
 GETNEXT ACTIVITY, Befehl [253](#)
 INQUIRE ACTIVITYID, Befehl [272](#)
 INQUIRE CONTAINER, Befehl [273](#)
 INQUIRE EVENT, Befehl [275](#)

ACTIVITYERR, Bedingung (*Forts.*)
 INQUIRE TIMER, Befehl [277](#)
 LINK ACTIVITY, Befehl [335](#)
 MOVE CONTAINER (BTS), Befehl [344](#)
 PUT CONTAINER (BTS), Befehl [354](#)
 RESET ACTIVITY, Befehl [448](#)
 RESUME, Befehl [454](#)
 RUN, Befehl [479](#)
 STARTBROWSE ACTIVITY, Befehl [591](#)
 STARTBROWSE CONTAINER, Befehl [592](#)
 STARTBROWSE EVENT, Befehl [595](#)
 SUSPEND (BTS), Befehl [598](#)

ACTIVITYID, Option
 ACQUIRE, Befehl [29](#)
 ASSIGN, Befehl [45](#)
 DEFINE ACTIVITY, Befehl [105](#)
 GETNEXT ACTIVITY, Befehl [252](#)
 GETNEXT PROCESS, Befehl [257](#)
 INQUIRE ACTIVITYID, Befehl [270](#)
 INQUIRE CONTAINER, Befehl [273](#)
 INQUIRE EVENT, Befehl [274](#)
 INQUIRE PROCESS, Befehl [276](#)
 INQUIRE TIMER, Befehl [277](#)
 STARTBROWSE ACTIVITY, Befehl [590](#)
 STARTBROWSE CONTAINER, Befehl [592](#)
 STARTBROWSE EVENT, Befehl [594](#)

ACTPARTN, Option
 SEND CONTROL, Befehl [501](#)
 SEND MAP, Befehl [507](#)
 SEND TEXT, Befehl [521](#)

ADD SUBEVENT, Befehl [31](#)

ADDRESS SET, Befehl [34](#)

ADDRESS, Befehl [32](#)

Adresse, Cursor [837](#)

ADS-Deskriptor [849](#)

ADS, Wert
 DFHMSD [871](#)

ADSL, Wert
 DFHMSD [871](#)

ADW1 [27](#)

AFTER, Option
 POST, Befehl [350](#)
 ROUTE, Befehl [473](#)
 START, Befehl [570](#)

AID-Zeichen (AID) [265](#)

AID, Option
 RECEIVE MAP MAPPINGDEV, Befehl [436](#)

Aktivitäten
 Löschung [129](#)
 Modi [271](#)
 Verarbeitungsstatus [271](#)

Aktivitätsbezogene Befehle
 ACQUIRE [28](#)
 CANCEL (BTS) [69](#)
 CHECK ACQPROCESS [75](#)
 CHECK ACTIVITY [77](#)
 DEFINE ACTIVITY [105](#)
 DEFINE PROCESS [113](#)
 DELETE ACTIVITY [129](#)
 INQUIRE ACTIVITYID [270](#)
 INQUIRE PROCESS [276](#)
 LINK ACQPROCESS [331](#)
 LINK ACTIVITY [334](#)
 RESET ACQPROCESS [446](#)

Aktivitätsbezogene Befehle (*Forts.*)

- RESET ACTIVITY [447](#)
- RESUME [453](#)
- RUN [476](#)
- STARTBROWSE ACTIVITY [590](#)
- SUSPEND (BTS) [597](#)
- Aktualisieren von Datensätzen
 - Batch Data Interchange [313](#)
 - Dateisteuerung [467](#)
- ALARM, Option
 - SEND CONTROL, Befehl [502](#)
 - SEND MAP MAPPINGDEV, Befehl [513](#)
 - SEND MAP, Befehl [507](#)
 - SEND TEXT NOEDIT, Befehl [529](#)
 - SEND TEXT, Befehl [521](#)
- ALARM, Wert
 - DFHMDI [862](#)
 - DFHMSD [871](#)
- ALIGNED, Attribut
 - PL/I [7](#)
- ALL, Option
 - SEND PAGE, Befehl [516](#)
- Allgemeiner Puffer, Ausgabe bei 2980 [427](#)
- ALLOCATE (APPC), Befehl [34](#)
- ALLOCATE (LUTYPE6.1), Befehl [37](#)
- ALLOCATE (MRO), Befehl [39](#)
- ALLOCERR, Bedingung
 - SPOOLCLOSE, Befehl [551](#)
 - SPOOLOPEN INPUT, Befehl [553](#)
 - SPOOLOPEN OUTPUT, Befehl [557](#)
 - SPOOLREAD, Befehl [560](#)
 - SPOOLWRITE, Befehl [563](#)
- ALTER, Option
 - QUERY SECURITY, Befehl [368](#)
- ALTERNATE, Option
 - CONVERSE (Nicht-z/OS Communications Server), Befehl [99](#)
 - CONVERSE (z/OS Communications Server), Befehl [93](#)
 - SEND (ohne z/OS Communications Server), Befehl [497](#)
 - SEND (z/OS Communications Server), Befehl [493](#)
 - SEND CONTROL, Befehl [502](#)
 - SEND MAP, Befehl [507](#)
 - SEND TEXT NOEDIT, Befehl [529](#)
 - SEND TEXT, Befehl [521](#)
- ALTSCRN, Operand
 - DFHPSD [882](#)
- ALTSCRNHT, Option
 - ASSIGN, Befehl [45](#)
- ALTSCRNWD, Option
 - ASSIGN, Befehl [45](#)
- AND, Option
 - DEFINE COMPOSITE EVENT, Befehl [108](#)
- Anhängen
 - Task starten [575](#)
- Anwendungsentwicklung [831](#)
- Anwendungsleistung überwachen [340](#)
- ANY, Option
 - FETCH CHILD, Befehl [194](#)
- ANYKEY, Option
 - HANDLE AID, Befehl [265](#)
- APLKYBD, Option
 - ASSIGN, Befehl [45](#)
- APLTEXT, Option
 - ASSIGN, Befehl [45](#)

- APPC-Basisdatenaustausch
 - Befehle [17](#)
- APPC-Dialog, formatfrei
 - abnormal beenden [287](#)
 - Attribute extrahieren [176](#)
 - Befehle [17](#)
 - Daten empfangen [415](#)
 - Daten senden [483](#)
 - Partner über Fehler informieren [303](#)
 - positive Antwort absetzen [292](#)
 - Richtungswechsel anfordern [318](#)
 - Sitzungen an CICS zurückgeben [203](#)
 - Übertragung von kumulierten Daten sicherstellen [629](#)
 - Werte aus Anschlussheader abrufen [182](#)
- APPC, logische Einheit
 - Dialog einleiten [81](#)
 - senden und empfangen [84](#)
 - Sitzung anfordern [34](#)
 - zugeordnete Sitzungen an CICS zurückgeben [203](#)
- APPEND, Option
 - PUT CONTAINER (CHANNEL), Befehl [356](#)
 - PUT64 CONTAINER, Befehl [360](#)
- APPLICATION, Option
 - INVOKE APPLICATION, Befehl [279](#)
- APPLID, Option
 - ASSIGN, Befehl [45](#)
- APPNOTFOUND, Bedingung
 - INVOKE APPLICATION, Befehl [280](#)
- Argumentwerte
 - Assemblersprache [7](#)
 - C [4](#)
 - COBOL [3](#)
 - PL/I [6](#)
- AS, Option
 - MOVE CONTAINER (BTS), Befehl [344](#)
 - MOVE CONTAINER (CHANNEL), Befehl [346](#)
- ASA, Option
 - SPOOLOPEN OUTPUT, Befehl [555](#)
- ASIS, Option
 - CONVERSE (Nicht-z/OS Communications Server), Befehl [99](#)
 - CONVERSE (z/OS Communications Server), Befehl [93](#)
 - RECEIVE (z/OS Communications Server), Befehl [421](#)
 - RECEIVE MAP, Befehl [433](#)
 - RECEIVE PARTN, Befehl [438](#)
 - RECEIVE, Befehl (ohne z/OS Communications Server) [429](#)
 - SEND (ohne z/OS Communications Server), Befehl [497](#)
- ASKIP, Wert
 - DFHMDF [851](#)
- ASKTIME, Befehl [41](#)
- ASRAINTRPT, Option
 - ASSIGN, Befehl [45](#)
- ASRAKEY, Option
 - ASSIGN, Befehl [45](#)
- ASRAPSW, Option
 - ASSIGN, Befehl [46](#)
- ASRAPSW16, Option
 - ASSIGN, Befehl [46](#)
- ASRAREGS, Option
 - ASSIGN, Befehl [46](#)
- ASRAREGS64, Option
 - ASSIGN, Befehl [46](#)
- ASRASPC, Option

- ASRASPC, Option (*Forts.*)
 - ASSIGN, Befehl [46](#)
- ASRASTG, Option
 - ASSIGN, Befehl [47](#)
- Assemblersprache
 - Argumentwerte [7](#)
 - konvertierter Code [13](#)
 - LENGTH, Option, Standard [8](#)
 - Programmexit [13](#)
 - Registerinhalt [13](#)
- ASSIGN, Befehl [45](#), [51–53](#)
- Asynchrone API-Befehle [18](#)
- Asynchroner Interrupt [833](#)
- ASYNCHRONOUS, Option
 - RUN, Befehl [479](#)
- AT, Option
 - DEFINE TIMER, Befehl [116](#)
 - DOCUMENT INSERT, Befehl [147](#)
 - POST, Befehl [350](#)
 - ROUTE, Befehl [473](#)
 - START, Befehl [570](#)
- ATTACHID, Option
 - BUILD ATTACH (LUTYPE6.1) Befehl [63](#)
 - BUILD ATTACH (MRO), Befehl [65](#)
 - CONVERSE (Nicht-z/OS Communications Server), Befehl [99](#)
 - CONVERSE (z/OS Communications Server), Befehl [93](#)
 - EXTRACT ATTACH (LUTYPE6.1), Befehl [170](#)
 - EXTRACT ATTACH (MRO), Befehl [173](#)
 - SEND (ohne z/OS Communications Server), Befehl [498](#)
 - SEND (z/OS Communications Server), Befehl [493](#)
- ATTRB, Operand
 - DFHMDf [851](#)
 - DFHPDI [881](#)
- Attribute
 - Steuerzeichenliste DFHBMSCA [841](#)
- Aus Warteschlange entfernen [140](#)
- Ausgabe in allgemeinen Puffer, 2980 [427](#)
- Ausgabesteuerung, Terminalsystem IBM 2980 für allgemeine Bankgeschäfte [427](#)
- Ausnahmebedingungen, Befehle zur Unterstützung [21](#)
- AUTHENTICATE, Option
 - EXTRACT TCPIP, Befehl [183](#)
 - WEB CONVERSE (Client), Befehl [648](#)
 - WEB SEND, Befehl (Client) [711](#)
- Authentifizierungsbefehle [18](#)
- AUTOPAGE, Option
 - SEND PAGE, Befehl [516](#)
- AUXILIARY, Option
 - WRITEQ TS, Befehl [744](#)

B

- BASE, Operand
 - DFHMSD [871](#)
- BASE, Wert
 - DFHMDf [851](#)
 - DFHMDI [862](#)
 - DFHMSD [871](#)
- Basic Mapping Support (BMS)
 - ADS-Deskriptor [849](#)
 - Befehle [18](#)
 - benutzerdefinierten Datenstrom senden [528](#)
 - BMS mit vollem Funktionsumfang

- Basic Mapping Support (BMS) (*Forts.*)
 - BMS mit vollem Funktionsumfang (*Forts.*)
 - PURGE MESSAGE [352](#)
 - Eingabedaten mit MAPPINGDEV zuordnen [435](#)
 - Eingabedaten zuordnen [432](#)
 - Eingabepartition bestimmen [437](#)
 - Felddefinitions makro [849](#), [851](#)
 - logische Nachricht abschließen [516](#)
 - logische Nachricht löschen [352](#)
 - logische Nachricht weiterleiten [472](#)
 - Maskendefinitionsmakro [849](#), [862](#)
 - Maskengruppendefinitionsmakro [849](#), [871](#)
 - minimale BMS
 - RECEIVE MAP MAPPINGDEV, Befehl [435](#)
 - RECEIVE MAP, Befehl [432](#)
 - SEND CONTROL, Befehl [500](#)
 - SEND MAP MAPPINGDEV, Befehl [513](#)
 - SEND MAP, Befehl [505](#)
 - Partitionsdefinitionsmakro [850](#), [881](#)
 - Partitionsgruppendifinitionsmakro [850](#), [882](#)
 - spezifische Konstanten [841](#)
 - standardmäßige BMS
 - RECEIVE MAP, Befehl [432](#)
 - RECEIVE PARTN, Befehl [437](#)
 - SEND CONTROL, Befehl [500](#)
 - SEND MAP, Befehl [505](#)
 - SEND PARTNSET [519](#)
 - SEND TEXT, Befehl [519](#)
 - vollständige BMS
 - RECEIVE MAP, Befehl [432](#)
 - RECEIVE PARTN, Befehl [437](#)
 - SEND CONTROL, Befehl [500](#)
 - SEND MAP, Befehl [505](#)
 - SEND PAGE [516](#)
 - SEND PARTNSET [519](#)
 - SEND TEXT MAPPED [526](#)
 - SEND TEXT NOEDIT [528](#)
 - SEND TEXT, Befehl [519](#)
 - zuvor zugeordnete Daten senden [526](#)
- Batch Data Interchange (BDI)
 - auf Funktionsabschluss warten [320](#)
 - Bedingungen [306](#)
 - Befehle [18](#)
 - Datei beenden [288](#), [296](#)
 - Daten an Ausgabeeinheit senden [316](#)
 - Datensatz aus Datei lesen [312](#)
 - Datensatz aus Datei löschen [300](#)
 - Datensatz in Datei aktualisieren [313](#)
 - Datensatz zu Datei hinzufügen [290](#)
 - nächste Datensatznummer anfordern [305](#)
- Beenden der Dateiverarbeitung
 - abnormal [288](#)
 - normal [296](#)
- Befehle
 - Format, Argumente [1](#)
 - Sicherheit [22](#)
 - Spool [22](#)
 - Steuerung des temporären Speichers [23](#)
 - TCP/IP [23](#)
 - Zeitplanung [22](#)
- Beispiele
 - Befehl READQ TS verwenden [414](#)
 - Befehl RELEASE verwenden [441](#)
 - Befehl RETRIEVE verwenden [457](#)

Beispiele (Forts.)

- Befehl REWRITE verwenden [472](#)
- Befehl START verwenden [565](#)
- Befehl WAIT EVENT verwenden [632](#)
- Befehl WAIT EXTERNAL verwenden [634](#)
- Befehl WAIT JOURNALNAME verwenden [637](#)
- Befehl WAITCICS verwenden [639](#)
- Befehl WRITE JOURNALNAME verwenden [737](#)
- Befehl WRITE verwenden [734](#)
- Befehl XCTL verwenden [763](#)
 - mit Befehl ADDRESS SET [34](#)
 - mit Befehl ASKTIME [41](#)
 - mit Befehl BIF DEEDIT [60](#)
 - mit Befehl CANCEL [27](#)
 - mit Befehl DELAY [118](#)
 - mit Befehl DELETE [121](#)
 - mit Befehl DEQ [142](#)
 - mit Befehl DUMP TRANSACTION [154](#)
 - mit Befehl ENQ [167](#)
 - mit Befehl ENTER TRACENUM [169](#)
 - mit Befehl FORMATTIME [202](#)
 - mit Befehl FREEMAIN [208](#)
 - mit Befehl GETMAIN [248](#), [252](#)
 - mit Befehl HANDLE ABEND [265](#)
 - mit Befehl HANDLE AID [267](#)
 - mit Befehl LINK [330](#)
 - mit Befehl LOAD [340](#)
 - mit Befehl MONITOR [342](#)
 - mit Befehl POST [352](#)
 - mit Befehl READ [386](#)
- READQ TD, Befehl verwenden [411](#)
- START ATTACH, Befehl verwenden [575](#)
- START BREXIT, Befehl verwenden [576](#)
- BELOW, Option
 - GETMAIN, Befehl [246](#)
- Benannte Zähler, Server, Befehle [22](#)
- Benannter Zähler
 - Abrage, benannter Zähler [364](#)
 - benannten Zähler definieren [109](#)
 - benannten Zähler löschen [134](#)
- Benannter Zählerserver, Befehl GET [240](#)
- Bezeichnung (Label), Argument, CICS-Befehlsformat [2](#)
- BIF DEEDIT, Befehl [60](#)
- BIF DIGEST, Befehl [61](#)
- Bildschirmeinheiten, Operationen
 - AID-Zeichen [837](#)
 - AID-Zeichenliste, DFHAID [846](#)
 - angezeigte Informationen kopieren [836](#)
 - Cursoradresse [837](#)
 - Drucken von angezeigten Informationen [836](#)
 - Eingabeoperation ohne Daten [837](#)
 - Entfernen aller ungeschützten Felder [837](#)
 - Standardliste für Attribute und Druckersteuerzeichen, DFHBMSCA [841](#)
 - Steuerung bei Empfang eines AID-Zeichens übergeben [265](#)
 - Terminal [835](#)
- BINARY, Option
 - DOCUMENT INSERT, Befehl [147](#)
- BLANK, Wert
 - DFHMDMF [851](#)
- BLINK, Wert
 - DFHMDMF [851](#)
 - DFHMDI [862](#)

BLINK, Wert (Forts.)

- DFHMSD [871](#)
- BLOCK, Wert
 - DFHMDI [862](#)
 - DFHMSD [871](#)
- BODYCHARSET, Option
 - WEB CONVERSE, Befehl [652](#)
 - WEB RECEIVE, Befehl (Client) [693](#)
 - WEB RECEIVE, Befehl (Server) [686](#)
- BOOKMARK, Option
 - DOCUMENT INSERT, Befehl [147](#)
- BOTTOM, Wert
 - DFHMDI [862](#)
- BRDATA, Option
 - START BREXIT, Befehl [576](#)
- BRDATALENGTH, Option
 - START BREXIT, Befehl [576](#)
- BREXIT, Option
 - START BREXIT, Befehl [576](#)
- BRIDGE, Option
 - ASSIGN, Befehl [47](#)
- Browseoperation
 - beenden [159](#)
 - starten [583](#)
 - vorherigen Datensatz lesen [398](#)
- BROWSETOKEN, Option
 - ENDBROWSE ACTIVITY, Befehl [161](#)
 - ENDBROWSE CONTAINER (CHANNEL), Befehl [162](#)
 - ENDBROWSE CONTAINER, Befehl [162](#)
 - ENDBROWSE EVENT, Befehl [163](#)
 - ENDBROWSE PROCESS, Befehl [164](#)
 - GETNEXT ACTIVITY, Befehl [252](#)
 - GETNEXT CONTAINER (CHANNEL), Befehl [255](#)
 - GETNEXT CONTAINER, Befehl [254](#)
 - GETNEXT EVENT, Befehl [255](#)
 - GETNEXT PROCESS, Befehl [257](#)
 - STARTBROWSE ACTIVITY, Befehl [591](#)
 - STARTBROWSE CONTAINER, Befehl [592](#), [594](#)
 - STARTBROWSE EVENT, Befehl [595](#)
 - STARTBROWSE PROCESS, Befehl [596](#)
- Browsingbefehle
 - ENDBROWSE ACTIVITY [161](#)
 - ENDBROWSE CONTAINER [162](#)
 - ENDBROWSE CONTAINER (CHANNEL) [162](#)
 - ENDBROWSE PROCESS [163](#)
 - ENDBROWSE TIMER [164](#)
 - GETNEXT ACTIVITY [252](#)
 - GETNEXT CONTAINER [253](#)
 - GETNEXT CONTAINER (CHANNEL) [254](#)
 - GETNEXT EVENT [255](#)
 - GETNEXT PROCESS [257](#)
 - GETNEXT TIMER [258](#)
 - INQUIRE ACTIVITYID [270](#)
 - INQUIRE CONTAINER [272](#)
 - INQUIRE EVENT [274](#)
 - INQUIRE PROCESS [276](#)
 - INQUIRE TIMER [277](#)
 - STARTBROWSE ACTIVITY [590](#)
 - STARTBROWSE CONTAINER [591](#)
 - STARTBROWSE CONTAINER (CHANNEL) [593](#)
 - STARTBROWSE PROCESS [595](#)
 - STARTBROWSE TIMER [596](#)
- BRT, Wert
 - DFHMDMF [851](#)

- Brücke (3270)
 - Task starten [576](#)
- BTRANS, Option
 - ASSIGN, Befehl [47](#)
- BTS-Befehle
 - ACQUIRE [28](#)
 - ADD SUBEVENT [31](#)
 - CANCEL (BTS) [69](#)
 - CHECK ACQPROCESS [75](#)
 - CHECK ACTIVITY [77](#)
 - CHECK TIMER [80](#)
 - DEFINE ACTIVITY [105](#)
 - DEFINE COMPOSITE EVENT [107](#)
 - DEFINE INPUT EVENT [112](#)
 - DEFINE PROCESS [113](#)
 - DEFINE TIMER [115](#)
 - DELETE ACTIVITY [129](#)
 - DELETE CONTAINER (BTS) [131](#)
 - DELETE EVENT [136](#)
 - DELETE TIMER [137](#)
 - ENDBROWSE ACTIVITY [161](#)
 - ENDBROWSE CONTAINER [162](#)
 - ENDBROWSE EVENT [163](#)
 - ENDBROWSE PROCESS [163](#)
 - ENDBROWSE TIMER [164](#)
 - FORCE TIMER [196](#)
 - GET CONTAINER (BTS) [233](#)
 - GETNEXT ACTIVITY [252](#)
 - GETNEXT CONTAINER [253](#)
 - GETNEXT EVENT [255](#)
 - GETNEXT PROCESS [257](#)
 - GETNEXT TIMER [258](#)
 - INQUIRE ACTIVITYID [270](#)
 - INQUIRE CONTAINER [272](#)
 - INQUIRE EVENT [274](#)
 - INQUIRE PROCESS [276](#)
 - INQUIRE TIMER [277](#)
 - LINK ACQPROCESS [331](#)
 - LINK ACTIVITY [334](#)
 - MOVE CONTAINER (BTS) [342](#)
 - PUT CONTAINER (BTS) [353](#)
 - REMOVE SUBEVENT [441](#)
 - RESET ACQPROCESS [446](#)
 - RESET ACTIVITY [447](#)
 - RESUME [453](#)
 - RETRIEVE REATTACH EVENT [458](#)
 - RETRIEVE SUBEVENT [459](#)
 - RUN [476](#)
 - STARTBROWSE ACTIVITY [590](#)
 - STARTBROWSE CONTAINER [591](#)
 - STARTBROWSE EVENT [594](#)
 - STARTBROWSE PROCESS [595](#)
 - STARTBROWSE TIMER [596](#)
 - SUSPEND (BTS) [597](#)
 - TEST EVENT [601](#)
- BUFFER, Option
 - GDS RECEIVE, Befehl [228](#)
 - RECEIVE (z/OS Communications Server), Befehl [422](#)
 - RECEIVE, Befehl (ohne z/OS Communications Server) [429](#)
- BUFSZE, Operand
 - DFHPDI [881](#)
- BUILD ATTACH (LUTYPE6.1), Befehl [62](#)
- BUILD ATTACH (MRO), Befehl [65](#)

- BUSY, Bedingung
 - DELETE COUNTER, Befehl [134](#)
- BYTEOFFSET, Option
 - GET CONTAINER (CHANNEL), Befehl [236](#)
 - GET64 CONTAINER, Befehl [260](#)

C

- C, Programmiersprache
 - ADDRESS COMMAREA [33](#)
 - ADDRESS EIB [33](#)
 - Argumentwerte [4](#)
 - konvertierter Code [12](#)
 - LENGTH, Option, Standard [5](#)
- CADDRLENGTH, Option
 - EXTRACT TCPIP, Befehl [183](#)
- CANCEL, Befehl [67](#)
- CANCEL, Befehl (BTS) [69](#)
- CANCEL, Option
 - ABEND, Befehl [27](#)
 - HANDLE ABEND, Befehl [264](#)
- CANCELLING, Modus einer Aktivität [271](#)
- CARD, Option
 - ISSUE ABORT, Befehl [289](#)
 - ISSUE END, Befehl [297](#)
 - ISSUE SEND, Befehl [316](#)
 - ISSUE WAIT, Befehl [321](#)
- CASE, Operand
 - DFHMD5 [851](#)
- CBIDERR, Bedingung
 - ALLOCATE (APPC), Befehl [37](#)
 - ALLOCATE (LUTYPE6.1), Befehl [38](#)
 - CONVERSE (Nicht-z/OS Communications Server), Befehl [102](#)
 - CONVERSE (z/OS Communications Server), Befehl [96](#)
 - EXTRACT ATTACH (LUTYPE6.1), Befehl [172](#)
 - EXTRACT ATTACH (MRO), Befehl [175](#)
 - SEND (ohne z/OS Communications Server), Befehl [500](#)
 - SEND (z/OS Communications Server), Befehl [495](#)
- CBUFF, Option
 - SEND (ohne z/OS Communications Server), Befehl [498](#)
- CCSID, Option
 - GET CONTAINER (CHANNEL), Befehl [236](#)
 - GET64 CONTAINER, Befehl [260](#)
- CCSIDERR, Bedingung
 - GET CONTAINER (CHANNEL), Befehl [239](#)
 - GET64 CONTAINER, Befehl [262](#)
 - PUT CONTAINER (CHANNEL), Befehl [358](#)
 - PUT64 CONTAINER, Befehl [362](#)
 - SOAPFAULT CREATE, Befehl [548](#)
 - WSACONTEXT BUILD, Befehl [750](#)
- CDT (Klassendeskiptortabelle)
 - Ressourcenlänge [369](#)
- CEE3250C [27](#)
- CEMSG [27](#)
- CERTIFICATE, Option
 - EXTRACT CERTIFICATE, Befehl [179](#)
 - WEB OPEN, Befehl [667](#)
- CHANGE PASSWORD, Befehl [73](#)
- CHANGE PHRASE, Befehl [71](#)
- CHANGE TASK, Befehl [75](#)
- CHANGED, Bedingung
 - DELETE, Befehl [121](#)
 - REWRITE, Befehl [469](#)

CHANGETIME, Option
 VERIFY PASSWORD, Befehl [619](#)
 VERIFY PHRASE, Befehl [623](#)
 CHANNEL, Option
 ASSIGN, Befehl [47](#)
 DELETE CHANNEL, Befehl [131](#)
 DELETE CONTAINER (CHANNEL), Befehl [133](#)
 FETCH CHILD, Befehl [194](#), [195](#)
 GET CONTAINER (CHANNEL) Befehl [236](#)
 GET64 CONTAINER, Befehl [260](#)
 INVOKE APPLICATION, Befehl [279](#)
 LINK, Befehl [324](#)
 MOVE CONTAINER (CHANNEL), Befehl [346](#)
 PUT CONTAINER (CHANNEL), Befehl [355](#)
 PUT64 CONTAINER, Befehl [359](#)
 QUERY CHANNEL, Befehl [363](#)
 RETURN, Befehl [461](#)
 RUN TRANSID, Befehl [481](#)
 START TRANSID (CHANNEL), Befehl [579](#)
 TRANSFORM XMLTODATA, Befehl [608](#)
 WEB CONVERSE, Befehl [648](#)
 WEB SEND, Befehl (Client) [711](#)
 WEB SEND, Befehl (Server) [701](#)
 XCTL, Befehl [760](#)
 CHANNELERR, Bedingung
 DELETE CHANNEL, Befehl [131](#)
 DELETE CONTAINER (CHANNEL), Befehl [133](#)
 GET CONTAINER (CHANNEL), Befehl [239](#)
 GET64 CONTAINER, Befehl [262](#)
 INVOKE APPLICATION, Befehl [281](#)
 LINK, Befehl [327](#)
 MOVE CONTAINER (CHANNEL), Befehl [346](#)
 PUT CONTAINER (CHANNEL), Befehl [358](#)
 PUT64 CONTAINER, Befehl [362](#)
 QUERY CHANNEL, Befehl [364](#)
 RETURN, Befehl [463](#)
 SIGNAL EVENT, Befehl [532](#)
 SOAPFAULT ADD, Befehl [545](#)
 SOAPFAULT CREATE, Befehl [549](#)
 SOAPFAULT DELETE, Befehl [550](#)
 START TRANSID (CHANNEL), Befehl [579](#)
 WEB CONVERSE, Befehl [656](#)
 WEB RECEIVE (Server), Befehl [690](#)
 WEB RECEIVE, Befehl (Client) [697](#)
 WEB SEND, Befehl (Client) [717](#)
 WEB SEND, Befehl (Server) [706](#)
 WSACONTEXT BUILD, Befehl [750](#)
 XCTL, Befehl [761](#)
 CHARACTERSET, Option
 DOCUMENT RETRIEVE, Befehl [151](#)
 WEB CONVERSE, Befehl [648](#)
 WEB READ FORMFIELD, Befehl [675](#)
 WEB RECEIVE, Befehl (Server) [687](#)
 WEB SEND, Befehl (Client) [711](#)
 WEB SEND, Befehl (Server) [701](#)
 WEB STARTBROWSE FORMFIELD, Befehl [720](#)
 CHARSZE, Operand
 DFHPDI [881](#)
 DFHPSD [882](#)
 CHECK ACQPROCESS, Befehl [75](#)
 CHECK ACTIVITY, Befehl [77](#)
 CHECK TIMER, Befehl [80](#)
 CHILD, Option
 FETCH CHILD, Befehl [195](#)
 CHILD, Option (*Forts.*)
 RUN TRANSID, Befehl [482](#)
 CHUNKING, Option
 WEB SEND, Befehl (Client) [712](#)
 WEB SEND, Befehl (Server) [702](#)
 CICS Business Transaction Services (BTS)
 Befehle [19](#)
 CICS Web Interface (CWI), Befehle
 DOCUMENT CREATE [142](#)
 DOCUMENT DELETE [146](#)
 DOCUMENT INSERT [146](#)
 DOCUMENT RETRIEVE [150](#)
 DOCUMENT SET [152](#)
 EXTRACT CERTIFICATE [178](#)
 CICS-Ereignisverarbeitungsbefehle
 SIGNAL EVENT [531](#)
 CICS-Webunterstützungsbefehle
 CONVERSE WEB [644](#)
 EXTRACT WEB [188](#)
 WEB CLOSE [642](#)
 WEB CONVERSE [644](#)
 WEB ENDBROWSE FORMFIELD [658](#)
 WEB ENDBROWSE HTTPHEADER [659](#)
 WEB ENDBROWSE QUERYPARM [660](#)
 WEB EXTRACT [660](#)
 WEB OPEN [666](#)
 WEB PARSE URL [671](#)
 WEB READ FORMFIELD [674](#)
 WEB READ HTTPHEADER [676](#)
 WEB READ QUERYPARM [678](#)
 WEB READNEXT FORMFIELD [680](#)
 WEB READNEXT HTTPHEADER [681](#)
 WEB READNEXT QUERYPARM [683](#)
 WEB RECEIVE [684](#)
 WEB RECEIVE (Client) [691](#)
 WEB RETRIEVE [698](#)
 WEB SEND (Client) [708](#)
 WEB SEND (Server) [699](#)
 WEB STARTBROWSE FORMFIELD [720](#)
 WEB STARTBROWSE HTTPHEADER [721](#)
 WEB STARTBROWSE QUERYPARM [722](#)
 WEB WRITE HTTPHEADER [724](#)
 CICS DATAKEY, Option
 GETMAIN, Befehl [247](#)
 GETMAIN64, Befehl [250](#)
 CIPHERS, Option
 WEB OPEN, Befehl [667](#)
 CLASS, Option
 SPOOL OPEN INPUT, Befehl [552](#)
 SPOOL OPEN OUTPUT, Befehl [556](#)
 CLEAR, Option
 HANDLE AID, Befehl [265](#)
 CLIENTADDR, Option
 EXTRACT TCPIP, Befehl [185](#)
 CLIENTADDRNU, Option
 EXTRACT TCPIP, Befehl [185](#)
 Clientanforderung
 Informationen extrahieren [178](#)
 CLIENTCONV, Option
 WEB CONVERSE, Befehl [654](#)
 WEB RECEIVE, Befehl (Client) [694](#)
 WEB SEND, Befehl (Client) [713](#)
 CLIENTNAME, Option
 EXTRACT TCPIP, Befehl [185](#)

CLNTADDR6NU, Option
 EXTRACT TCPIP, Befehl [185](#)
 CLNTCODEPAGE, Option
 DOCUMENT RETRIEVE, Befehl [151](#)
 WEB READ FORMFIELD, Befehl [675](#)
 WEB RECEIVE, Befehl (Server) [687](#)
 WEB SEND, Befehl (Server) [702](#)
 WEB STARTBROWSE FORMFIELD, Befehl [720](#)
 CLNTIPFAMILY, Option
 EXTRACT TCPIP, Befehl [185](#)
 CLOSESTATUS, Option
 WEB CONVERSE, Befehl [648](#)
 WEB SEND, Befehl (Client) [712](#)
 WEB SEND, Befehl (Server) [702](#)
 CLRPARTN, Option
 HANDLE AID, Befehl [265](#)
 CMDSEC, Option
 ASSIGN, Befehl [47](#)
 CNAMELENGTH, Option
 EXTRACT TCPIP, Befehl [185](#)
 CNOTCOMPL, Option
 SEND (ohne z/OS Communications Server), Befehl [498](#)
 SEND (z/OS Communications Server), Befehl [493](#)
 COBOL
 Argumentwerte [3](#)
 konvertierter Code [11](#)
 CODEPAGE, Option
 WEB OPEN, Befehl [667](#)
 CODEPAGEERR, Bedingung
 GET CONTAINER (CHANNEL), Befehl [239](#)
 GET64 CONTAINER, Befehl [262](#)
 PUT CONTAINER (CHANNEL), Befehl [358](#)
 PUT64 CONTAINER, Befehl [362](#)
 WSACONTEXT BUILD, Befehl [751](#)
 COLOR, Operand
 DFHMDI [851](#)
 DFHMDI [862](#)
 DFHMSD [871](#)
 COLOR, Option
 ASSIGN, Befehl [48](#)
 COLUMN, Operand
 DFHMDI [862](#)
 COMMAREA-Größen über 32 K (Kanäle)
 Option CHANNEL des Befehls RETURN [460](#)
 COMMAREA, Option
 ADDRESS, Befehl [33](#)
 INVOKE APPLICATION, Befehl [279](#)
 LINK, Befehl [324](#)
 RETURN, Befehl [462](#)
 XCTL, Befehl [761](#)
 Common Programming Interface Communications (CPI Communications) [839](#)
 COMMONNAME, Option
 EXTRACT CERTIFICATE, Befehl [179](#)
 COMMONNAMLEN, Option
 EXTRACT CERTIFICATE, Befehl [180](#)
 COMPAREMAX, Option
 GET COUNTER, Befehl [240](#)
 GET DCOUNTER, Befehl [240](#)
 UPDATE COUNTER, Befehl [615](#)
 UPDATE DCOUNTER, Befehl [615](#)
 COMPAREMIN, Option
 GET COUNTER, Befehl [240](#)
 GET DCOUNTER, Befehl [240](#)
 COMPAREMIN, Option (*Forts.*)
 UPDATE COUNTER, Befehl [615](#)
 UPDATE DCOUNTER, Befehl [615](#)
 COMPLETE, Modus einer Aktivität [271](#)
 COMPLETE, Option
 DUMP TRANSACTION, Befehl [155](#)
 COMPOSITE, Option
 GETNEXT EVENT, Befehl [256](#)
 INQUIRE EVENT, Befehl [274](#)
 COMPSTATUS, Option
 CHECK ACQPROCESS, Befehl [76](#)
 CHECK ACTIVITY, Befehl [78](#)
 INQUIRE ACTIVITYID, Befehl [270](#)
 CONFIRM, Option
 GDS SEND, Befehl [230](#)
 SEND (z/OS Communications Server), Befehl [493](#)
 CONNECT PROCESS, Befehl [81](#)
 CONSISTENT, Option
 READ, Befehl [377](#)
 READNEXT, Befehl [390](#)
 READPREV, Befehl [401](#)
 CONSOLE, Option
 ISSUE ABORT, Befehl [289](#)
 ISSUE END, Befehl [297](#)
 ISSUE SEND, Befehl [316](#)
 ISSUE WAIT, Befehl [321](#)
 CONTAINER, Option
 DELETE CONTAINER (BTS), Befehl [132](#)
 DELETE CONTAINER (CHANNEL), Befehl [133](#)
 GET CONTAINER (BTS), Befehl [234](#)
 GET CONTAINER (CHANNEL), Befehl [237](#)
 GET64 CONTAINER, Befehl [260](#)
 GETNEXT CONTAINER (CHANNEL), Befehl [255](#)
 GETNEXT CONTAINER, Befehl [254](#)
 INQUIRE CONTAINER, Befehl [273](#)
 MOVE CONTAINER (BTS), Befehl [344](#)
 MOVE CONTAINER (CHANNEL), Befehl [346](#)
 PUT CONTAINER (BTS), Befehl [354](#)
 PUT CONTAINER (CHANNEL), Befehl [357](#)
 PUT64 CONTAINER, Befehl [361](#)
 WEB CONVERSE, Befehl [648](#)
 WEB SEND, Befehl (Client) [713](#)
 WEB SEND, Befehl (Server) [703](#)
 Containerbefehle
 DELETE CHANNEL [130](#)
 DELETE CONTAINER (BTS) [131](#)
 DELETE CONTAINER (CHANNEL) [133](#)
 ENDBROWSE CONTAINER [162](#)
 ENDBROWSE CONTAINER (CHANNEL) [162](#)
 GET CONTAINER (BTS) [233](#)
 GET CONTAINER (CHANNEL) [236](#)
 GET64 CONTAINER [259](#)
 GETNEXT CONTAINER [253](#)
 GETNEXT CONTAINER (CHANNEL) [254](#)
 INQUIRE CONTAINER [272](#)
 MOVE CONTAINER (BTS) [342](#)
 MOVE CONTAINER (CHANNEL) [345](#)
 PUT CONTAINER (BTS) [353](#)
 PUT CONTAINER (CHANNEL) [355](#)
 PUT64 CONTAINER [359](#)
 QUERY CHANNEL [363](#)
 STARTBROWSE CONTAINER [591](#)
 STARTBROWSE CONTAINER (CHANNEL) [593](#)
 CONTAINERCNT, Option

CONTAINERCNT, Option (*Forts.*)
 QUERY CHANNEL, Befehl [364](#)

CONTAINERERR, Bedingung
 DELETE CONTAINER (BTS), Befehl [132](#)
 DELETE CONTAINER (CHANNEL), Befehl [133](#)
 GET CONTAINER (BTS), Befehl [235](#)
 GET CONTAINER (CHANNEL), Befehl [239](#)
 GET64 CONTAINER, Befehl [263](#)
 INQUIRE CONTAINER, Befehl [273](#)
 MOVE CONTAINER (BTS), Befehl [344](#)
 MOVE CONTAINER (CHANNEL), Befehl [346](#)
 PUT CONTAINER (BTS), Befehl [355](#)
 PUT CONTAINER (CHANNEL), Befehl [358](#)
 PUT64 CONTAINER, Befehl [362](#)
 WEB CONVERSE, Befehl [656](#)
 WEB RECEIVE (Server), Befehl [690](#)
 WEB RECEIVE, Befehl (Client) [697](#)
 WEB SEND, Befehl (Client) [717](#)
 WEB SEND, Befehl (Server) [706](#)

CONTROL, Option
 QUERY SECURITY, Befehl [368](#)

CONVDATA, Option
 GDS CONNECT PROCESS, Befehl [215](#)
 GDS EXTRACT ATTRIBUTES, Befehl [217](#)
 GDS FREE, Befehl [220](#)
 GDS ISSUE ABEND, Befehl [221](#)
 GDS ISSUE CONFIRMATION, Befehl [222](#)
 GDS ISSUE ERROR, Befehl [224](#)
 GDS ISSUE PREPARE, Befehl [225](#)
 GDS ISSUE SIGNAL, Befehl [226](#)
 GDS RECEIVE, Befehl [228](#)
 GDS SEND, Befehl [230](#)
 GDS WAIT, Befehl [232](#)

CONVERSE (2260), Befehl [99](#)
 CONVERSE (3270 logisch), Befehl [87](#)
 CONVERSE (3600-3601), Befehl [87](#)
 CONVERSE (3600-3614), Befehl [88](#)
 CONVERSE (3650-3270), Befehl [89](#)
 CONVERSE (3650-3653), Befehl [90](#)
 CONVERSE (3650-3680), Befehl [90](#)
 CONVERSE (3650, Interpreter), Befehl [89](#)
 CONVERSE (3767), Befehl [91](#)
 CONVERSE (3770), Befehl [91](#)
 CONVERSE (3790, 3270-Anzeige), Befehl [92](#)
 CONVERSE (3790, mit vollem Funktionsumfang oder Anfrage), Befehl [92](#)
 CONVERSE (APPC), Befehl [84](#)
 CONVERSE (LUTYPE2/LUTYPE3), Befehl [84](#)
 CONVERSE (LUTYPE4), Befehl [85](#)
 CONVERSE (LUTYPE6.1), Befehl [86](#)
 CONVERSE (MRO), Befehl [98](#)
 CONVERSE (Nicht-z/OS Communications Server-Standard), Befehl [98](#)
 CONVERSE (SCS), Befehl [86](#)
 CONVERSE (Standard), Befehl [84](#)
 CONVERSE WEB, Befehl [644](#)
 CONVERSE, Option
 ISSUE LOAD, Befehl [305](#)

CONVERTST, Option
 GET CONTAINER (CHANNEL), Befehl [237](#)
 GET64 CONTAINER, Befehl [260](#)

CONVERTTIME, Befehl [103](#)

CONVID, Option
 CONNECT PROCESS, Befehl [81](#)

CONVID, Option (*Forts.*)
 CONVERSE (Nicht-z/OS Communications Server), Befehl [99](#)
 CONVERSE (z/OS Communications Server), Befehl [93](#)
 EXTRACT ATTACH (LUTYPE6.1), Befehl [170](#)
 EXTRACT ATTACH (MRO), Befehl [173](#)
 EXTRACT ATTRIBUTES (APPC), Befehl [176](#)
 EXTRACT ATTRIBUTES (MRO), Befehl [177](#)
 EXTRACT PROCESS, Befehl [182](#)
 FREE (APPC), Befehl [203](#)
 FREE (LUTYPE6.1), Befehl [204](#)
 FREE (MRO), Befehl [205](#)
 GDS ALLOCATE, Befehl [211](#)
 GDS CONNECT PROCESS, Befehl [215](#)
 GDS EXTRACT ATTRIBUTES, Befehl [217](#)
 GDS EXTRACT PROCESS, Befehl [219](#)
 GDS FREE, Befehl [220](#)
 GDS ISSUE ABEND, Befehl [221](#)
 GDS ISSUE CONFIRMATION, Befehl [223](#)
 GDS ISSUE ERROR, Befehl [224](#)
 GDS ISSUE PREPARE, Befehl [225](#)
 GDS ISSUE SIGNAL, Befehl [226](#)
 GDS RECEIVE, Befehl [228](#)
 GDS SEND, Befehl [230](#)
 GDS WAIT, Befehl [232](#)
 ISSUE ABEND, Befehl [287](#)
 ISSUE CONFIRMATION, Befehl [292](#)
 ISSUE ERROR, Befehl [303](#)
 ISSUE PREPARE, Befehl [309](#)
 ISSUE SIGNAL (APPC), Befehl [318](#)
 ISSUE SIGNAL (LUTYPE6.1), Befehl [320](#)
 POINT, Befehl [347](#)
 RECEIVE (z/OS Communications Server), Befehl [422](#)
 SEND (z/OS Communications Server), Befehl [493](#)
 WAIT CONVID, Befehl [629](#)
 WAIT TERMINAL, Befehl [638](#)

Copybooks
 DFHAID [846](#)
 DFHBMSCA [841](#)
 DFHEIBLK [13](#)
 DFHMSRCA [845](#)

COUNTER, Option
 DEFINE COUNTER, Befehl [109](#)
 DEFINE DCOUNTER, Befehl [109](#)
 DELETE COUNTER, Befehl [134](#)
 GET COUNTER, Befehl [240](#)
 QUERY COUNTER, Befehl [364](#)
 REWIND COUNTER, Befehl [464](#)
 UPDATE COUNTER, Befehl [615](#)

COUNTRY, Option
 EXTRACT CERTIFICATE, Befehl [180](#)

COUNTRYLEN, Option
 EXTRACT CERTIFICATE, Befehl [180](#)

CPI Communications (SAA) [839](#)

CTLCHAR, Option
 CONVERSE (Nicht-z/OS Communications Server), Befehl [99](#)
 CONVERSE (z/OS Communications Server), Befehl [94](#)
 ISSUE COPY (3270, logisch), Befehl [294](#)
 SEND (ohne z/OS Communications Server), Befehl [498](#)
 SEND (z/OS Communications Server), Befehl [493](#)

CTRL, Operand
 DFHMDI [862](#)
 DFHMDS [871](#)

CURRENT, Option
 SEND PAGE, Befehl [517](#)

CURSLOC, Operand
 DFHMDI [862](#)
 DFHMSD [871](#)

CURSOR, Option
 RECEIVE MAP MAPPINGDEV, Befehl [436](#)
 SEND CONTROL, Befehl [502](#)
 SEND MAP MAPPINGDEV, Befehl [513](#)
 SEND MAP, Befehl [507](#)
 SEND TEXT, Befehl [521](#)

Cursoradresse [837](#)

Cursorposition
 Terminalsteuerung [837](#)

CVDA (Datenbereich für CICS-Werte)
 Befehlsformat [2](#)
 in numerischer und alphabetischer Reihenfolge [769](#)

CVDA-Optionen
 ACTION
 WRITE OPERATOR, Befehl [739](#)

ALTER
 QUERY SECURITY, Befehl [368](#)

ASRAKEY
 ASSIGN, Befehl [45](#)

ASRASPC
 ASSIGN, Befehl [46](#)

CONTROL
 QUERY SECURITY, Befehl [368](#)

LOGMESSAGE
 QUERY SECURITY, Befehl [368](#)

MAXLIFETIME
 DEQ [142](#)
 ENQ [166](#)

PURGEABILITY
 WAIT EXTERNAL [633](#)
 WAITCICS [640](#)

READ
 QUERY SECURITY, Befehl [368](#)

STATE
 ALLOCATE (APPC) [36](#)
 ALLOCATE (MRO) [40](#)
 CONNECT PROCESS [82](#)
 EXTRACT ATTRIBUTES (APPC) [176](#)
 EXTRACT ATTRIBUTES (MRO) [177](#)
 FREE (APPC) [203](#)
 FREE (MRO) [205](#)
 GDS ALLOCATE [212](#)
 GDS CONNECT PROCESS [216](#)
 GDS EXTRACT ATTRIBUTES [217](#)
 GDS FREE [220](#)
 GDS ISSUE ABEND [221](#)
 GDS ISSUE CONFIRMATION [223](#)
 GDS ISSUE ERROR [224](#)
 GDS ISSUE PREPARE [225](#)
 GDS ISSUE SIGNAL [227](#)
 GDS RECEIVE [229](#)
 GDS SEND [231](#)
 GDS WAIT [232](#)
 ISSUE ABEND, Befehl [287](#)
 ISSUE CONFIRMATION [292](#)
 ISSUE ERROR [304](#)
 ISSUE PREPARE [309](#)
 ISSUE SIGNAL (APPC) [318](#)
 WAIT CONVID, Befehl [630](#)

CVDA-Optionen (Forts.)
 UPDATE
 QUERY SECURITY, Befehl [372](#)

CVDA-Werte
 ALLOCATED
 ALLOCATE (APPC) [36](#)
 ALLOCATE (MRO) [40](#)
 CONNECT PROCESS [82](#)
 EXTRACT ATTRIBUTES (APPC) [176](#)
 EXTRACT ATTRIBUTES (MRO) [177](#)
 FREE (APPC) [203](#)
 FREE (MRO) [205](#)
 GDS ALLOCATE [212](#)
 GDS CONNECT PROCESS [216](#)
 GDS EXTRACT ATTRIBUTES [217](#)
 GDS FREE [220](#)
 GDS ISSUE ABEND [221](#)
 GDS ISSUE CONFIRMATION [223](#)
 GDS ISSUE ERROR [224](#)
 GDS ISSUE PREPARE [225](#)
 GDS ISSUE SIGNAL [227](#)
 GDS RECEIVE [229](#)
 GDS SEND [231](#)
 GDS WAIT [232](#)
 ISSUE ABEND, Befehl [287](#)
 ISSUE CONFIRMATION [292](#)
 ISSUE ERROR [304](#)
 ISSUE PREPARE [309](#)
 ISSUE SIGNAL (APPC) [318](#)
 RECEIVE (z/OS Communications Server), Befehl [423](#)
 RECEIVE, Befehl (MRO) [431](#)
 SEND (ohne z/OS Communications Server), Befehl [499](#)
 SEND (z/OS Communications Server), Befehl [494](#)
 WAIT CONVID, Befehl [630](#)

ASSERTED
 EXTRACT TCPIP, Befehl [183](#)

ATTLAWARE
 EXTRACT TCPIP, Befehl [183](#), [187](#)

AUTOAUTH
 EXTRACT TCPIP, Befehl [183](#)

AUTOREGISTER
 EXTRACT TCPIP, Befehl [183](#)

BASE64
 BIF DIGEST, Befehl [61](#)

BASSPACE
 ASSIGN, Befehl [47](#)

BASICAUTH
 EXTRACT TCPIP, Befehl [183](#)
 WEB CONVERSE (Client), Befehl [648](#)
 WEB SEND, Befehl (Client) [711](#)

BINARY
 BIF DIGEST, Befehl [61](#)

CERTIFICAUTH
 EXTRACT TCPIP, Befehl [183](#)

CHUNKEND
 WEB SEND, Befehl (Client) [712](#)
 WEB SEND, Befehl (Server) [702](#)

CHUNKNO
 WEB SEND, Befehl (Client) [712](#)
 WEB SEND, Befehl (Server) [702](#)

CHUNKYES
 WEB SEND, Befehl (Client) [712](#)

CVDA-Werte (Forts.)

CHUNKYES (Forts.)

WEB SEND, Befehl (Server) [702](#)

CICSEXECKEY

ASSIGN, Befehl [46](#)

CLICONVERT

Befehl WEB RECEIVE (Client) [694](#)

WEB CONVERSE, Befehl [655](#)

WEB SEND, Befehl (Client) [713](#)

CLIENTAUTH

EXTRACT TCPIP, Befehl [187](#)

CLOSE

WEB CONVERSE, Befehl [648](#)

WEB SEND, Befehl (Client) [712](#)

WEB SEND, Befehl (Server) [702](#)

CONFFREE

CONNECT PROCESS [82](#)

EXTRACT ATTRIBUTES (APPC) [176](#)

FREE (APPC) [203](#)

GDS ALLOCATE [212](#)

GDS CONNECT PROCESS [216](#)

GDS EXTRACT ATTRIBUTES [217](#)

GDS FREE [220](#)

GDS ISSUE ABEND [221](#)

GDS ISSUE CONFIRMATION [223](#)

GDS ISSUE ERROR [224](#)

GDS ISSUE PREPARE [225](#)

GDS ISSUE SIGNAL [227](#)

GDS RECEIVE [229](#)

GDS SEND [231](#)

GDS WAIT [232](#)

ISSUE ABEND, Befehl [287](#)

ISSUE CONFIRMATION [292](#)

ISSUE ERROR [304](#)

ISSUE PREPARE [309](#)

ISSUE SIGNAL (APPC) [318](#)

RECEIVE (z/OS Communications Server), Befehl [423](#)

SEND (z/OS Communications Server), Befehl [494](#)

WAIT CONVID, Befehl [630](#)

CONFRECEIVE

CONNECT PROCESS [82](#)

EXTRACT ATTRIBUTES (APPC) [176](#)

FREE (APPC) [203](#)

GDS ALLOCATE [212](#)

GDS CONNECT PROCESS [216](#)

GDS EXTRACT ATTRIBUTES [217](#)

GDS FREE [220](#)

GDS ISSUE ABEND [222](#)

GDS ISSUE CONFIRMATION [223](#)

GDS ISSUE ERROR [224](#)

GDS ISSUE PREPARE [225](#)

GDS ISSUE SIGNAL [227](#)

GDS RECEIVE [229](#)

GDS SEND [231](#)

GDS WAIT [232](#)

ISSUE ABEND, Befehl [287](#)

ISSUE CONFIRMATION [293](#)

ISSUE ERROR [304](#)

ISSUE PREPARE [309](#)

ISSUE SIGNAL (APPC) [318](#)

RECEIVE (z/OS Communications Server), Befehl [423](#)

SEND (z/OS Communications Server), Befehl [494](#)

CVDA-Werte (Forts.)

CONFRECEIVE (Forts.)

WAIT CONVID, Befehl [630](#)

CONFSEND

CONNECT PROCESS [82](#)

EXTRACT ATTRIBUTES (APPC) [176](#)

FREE (APPC) [203](#)

GDS ALLOCATE [212](#)

GDS CONNECT PROCESS [216](#)

GDS EXTRACT ATTRIBUTES [217](#)

GDS FREE [220](#)

GDS ISSUE ABEND [222](#)

GDS ISSUE CONFIRMATION [223](#)

GDS ISSUE ERROR [224](#)

GDS ISSUE PREPARE [225](#)

GDS ISSUE SIGNAL [227](#)

GDS RECEIVE [229](#)

GDS SEND [231](#)

GDS WAIT [232](#)

ISSUE ABEND, Befehl [287](#)

ISSUE CONFIRMATION [293](#)

ISSUE ERROR [304](#)

ISSUE PREPARE [309](#)

ISSUE SIGNAL (APPC) [318](#)

RECEIVE (z/OS Communications Server), Befehl [423](#)

SEND (z/OS Communications Server), Befehl [494](#)

WAIT CONVID, Befehl [630](#)

CRITICAL

WRITE OPERATOR, Befehl [739](#)

DELETE

WEB CONVERSE, Befehl [650](#), [715](#)

DOCDELETE

WEB CONVERSE, Befehl [649](#)

WEB SEND, Befehl (Client) [714](#)

WEB SEND, Befehl (Server) [703](#)

EVENTUAL

WEB SEND, Befehl (Server) [701](#)

WRITE OPERATOR, Befehl [739](#)

EXPECT

WEB CONVERSE, Befehl [647](#)

WEB SEND, Befehl (Client) [711](#)

FREE

CONNECT PROCESS [82](#)

EXTRACT ATTRIBUTES (APPC) [176](#)

EXTRACT ATTRIBUTES (MRO) [177](#)

FREE (APPC) [203](#)

FREE (MRO) [205](#)

GDS ALLOCATE [212](#)

GDS CONNECT PROCESS [216](#)

GDS EXTRACT ATTRIBUTES [217](#)

GDS FREE [220](#)

GDS ISSUE ABEND [222](#)

GDS ISSUE CONFIRMATION [223](#)

GDS ISSUE ERROR [224](#)

GDS ISSUE PREPARE [225](#)

GDS ISSUE SIGNAL [227](#)

GDS RECEIVE [229](#)

GDS SEND [231](#)

GDS WAIT [232](#)

ISSUE ABEND, Befehl [287](#)

ISSUE CONFIRMATION [293](#)

ISSUE ERROR [304](#)

ISSUE PREPARE [309](#)

CVDA-Werte (Forts.)

FREE (Forts.)

ISSUE SIGNAL (APPC) [318](#)
 RECEIVE (z/OS Communications Server), Befehl [423](#)
 RECEIVE, Befehl (MRO) [431](#)
 SEND (ohne z/OS Communications Server), Befehl [499](#)
 SEND (z/OS Communications Server), Befehl [494](#)
 WAIT CONVID, Befehl [630](#)

GET

WEB CONVERSE, Befehl [650](#)
 WEB SEND, Befehl (Client) [715](#)

HEAD

WEB CONVERSE, Befehl [650](#)
 WEB SEND, Befehl (Client) [715](#)

HEX

BIF DIGEST, Befehl [61](#)

HOSTNAME

WEB EXTRACT oder EXTRACT WEB, Befehl [190](#),
[662](#)
 WEB PARSE URL, Befehl [673](#)

HTTP

WEB EXTRACT oder EXTRACT WEB, Befehl [191](#),
[664](#)
 WEB OPEN, Befehl [669](#)

HTTPNO

WEB EXTRACT oder EXTRACT WEB, Befehl [191](#),
[664](#)
 WEB RECEIVE (Server), Befehl [690](#)

HTTPS

WEB EXTRACT oder EXTRACT WEB, Befehl [191](#),
[664](#)
 WEB OPEN, Befehl [669](#)

HTTPYES

WEB EXTRACT oder EXTRACT WEB, Befehl [191](#),
[664](#)
 WEB RECEIVE (Server), Befehl [690](#)

IMMEDIATE

WEB SEND, Befehl (Server) [701](#)
 WRITE OPERATOR, Befehl [739](#)

IPV4

EXTRACT TCPIP, Befehl [185](#), [186](#)
 WEB EXTRACT oder EXTRACT WEB, Befehl [190](#),
[662](#)
 WEB PARSE URL, Befehl [673](#)

IPV6

EXTRACT TCPIP, Befehl [185](#), [186](#)
 WEB EXTRACT oder EXTRACT WEB, Befehl [190](#),
[662](#)
 WEB PARSE URL, Befehl [673](#)

NOAUTHENTIC

EXTRACT TCPIP, Befehl [183](#)

NOCLICONVERT

Befehl WEB RECEIVE (Client) [694](#)
 WEB CONVERSE, Befehl [655](#)
 WEB SEND, Befehl (Client) [713](#)

NOCLOSE

WEB CONVERSE, Befehl [648](#)
 WEB SEND, Befehl (Client) [713](#)
 WEB SEND, Befehl (Server) [703](#)

NOCONVERT

GET CONTAINER (CHANNEL), Befehl [237](#)
 GET64 CONTAINER, Befehl [260](#)

CVDA-Werte (Forts.)

NODOCDELETE

WEB CONVERSE, Befehl [649](#)
 WEB SEND, Befehl (Client) [714](#)
 WEB SEND, Befehl (Server) [703](#)

NOINCONVERT

WEB CONVERSE, Befehl [655](#)

NONCICS

ASSIGN, Befehl [46](#)

NONE

WEB CONVERSE (Client), Befehl [648](#)
 WEB SEND, Befehl (Client) [711](#)

NOOUTCONVERT

WEB CONVERSE, Befehl [655](#)

NOSRVCONVERT

WEB RECEIVE (Server), Befehl [689](#)
 WEB SEND, Befehl (Server) [705](#)

NOSSL

EXTRACT TCPIP, Befehl [187](#)

NOTAPPLIC

ASSIGN, Befehl [46](#), [47](#)
 EXTRACT TCPIP, Befehl [185](#), [186](#)
 WEB EXTRACT oder EXTRACT WEB, Befehl [190](#),
[662](#)

NOTPURGEABLE

WAIT EXTERNAL [633](#)
 WAITCICS [640](#)

NOTSUPPORTED

EXTRACT TCPIP, Befehl [183](#)

OPTIONS

WEB CONVERSE, Befehl [650](#)
 WEB SEND, Befehl (Client) [715](#)

PATCH

WEB SEND, Befehl (Client) [650](#), [715](#)

PENDFREE

CONNECT PROCESS [82](#)
 EXTRACT ATTRIBUTES (APPC) [176](#)
 EXTRACT ATTRIBUTES (MRO) [177](#)
 FREE (APPC) [203](#)
 FREE (MRO) [206](#)
 GDS ALLOCATE [212](#)
 GDS CONNECT PROCESS [216](#)
 GDS EXTRACT ATTRIBUTES [217](#)
 GDS FREE [220](#)
 GDS ISSUE ABEND [222](#)
 GDS ISSUE CONFIRMATION [223](#)
 GDS ISSUE ERROR [224](#)
 GDS ISSUE PREPARE [225](#)
 GDS ISSUE SIGNAL [227](#)
 GDS RECEIVE [229](#)
 GDS SEND [231](#)
 GDS WAIT [232](#)
 ISSUE ABEND, Befehl [287](#)
 ISSUE CONFIRMATION [293](#)
 ISSUE ERROR [304](#)
 ISSUE PREPARE [309](#)
 ISSUE SIGNAL (APPC) [318](#)
 RECEIVE (z/OS Communications Server), Befehl [423](#)
 RECEIVE, Befehl (MRO) [431](#)
 SEND (ohne z/OS Communications Server), Befehl [499](#)
 SEND (z/OS Communications Server), Befehl [494](#)
 WAIT CONVID, Befehl [630](#)

CVDA-Werte (Forts.)

PENDRECEIVE

CONNECT PROCESS [82](#)
 EXTRACT ATTRIBUTES (APPC) [176](#)
 FREE (APPC) [203](#)
 GDS ALLOCATE [212](#)
 GDS CONNECT PROCESS [216](#)
 GDS EXTRACT ATTRIBUTES [217](#)
 GDS FREE [220](#)
 GDS ISSUE ABEND [222](#)
 GDS ISSUE CONFIRMATION [223](#)
 GDS ISSUE ERROR [224](#)
 GDS ISSUE PREPARE [225](#)
 GDS ISSUE SIGNAL [227](#)
 GDS RECEIVE [229](#)
 GDS SEND [231](#)
 GDS WAIT [232](#)
 ISSUE ABEND, Befehl [287](#)
 ISSUE CONFIRMATION [293](#)
 ISSUE ERROR [304](#)
 ISSUE PREPARE [309](#)
 ISSUE SIGNAL (APPC) [318](#)
 RECEIVE (z/OS Communications Server), Befehl [423](#)
 SEND (z/OS Communications Server), Befehl [494](#)
 WAIT CONVID, Befehl [630](#)

POST

WEB CONVERSE, Befehl [650](#)
 WEB SEND, Befehl (Client) [715](#)

PURGEABLE

WAIT EXTERNAL [633](#)
 WAITCICS [640](#)

PUT

WEB CONVERSE, Befehl [650](#)
 WEB SEND, Befehl (Client) [715](#)

RECEIVE

CONNECT PROCESS [82](#)
 CONVERSE, Befehl (Nicht-z/OS Communications Server) [101](#)
 EXTRACT ATTRIBUTES (APPC) [176](#)
 EXTRACT ATTRIBUTES (MRO) [177](#)
 FREE (APPC) [203](#)
 FREE (MRO) [206](#)
 GDS ALLOCATE [212](#)
 GDS CONNECT PROCESS [216](#)
 GDS EXTRACT ATTRIBUTES [218](#)
 GDS FREE [220](#)
 GDS ISSUE ABEND [222](#)
 GDS ISSUE CONFIRMATION [223](#)
 GDS ISSUE ERROR [224](#)
 GDS ISSUE PREPARE [226](#)
 GDS ISSUE SIGNAL [227](#)
 GDS RECEIVE [229](#)
 GDS SEND [231](#)
 GDS WAIT [233](#)
 ISSUE ABEND, Befehl [287](#)
 ISSUE CONFIRMATION [293](#)
 ISSUE ERROR [304](#)
 ISSUE PREPARE [309](#)
 ISSUE SIGNAL (APPC) [318](#)
 RECEIVE (z/OS Communications Server), Befehl [423](#)
 RECEIVE, Befehl (MRO) [431](#)

CVDA-Werte (Forts.)

RECEIVE (Forts.)

SEND (ohne z/OS Communications Server), Befehl [499](#)
 SEND (z/OS Communications Server), Befehl [494](#)
 WAIT CONVID, Befehl [630](#)

REQUIRED

EXTRACT TCPIP, Befehl [183](#)

RFC1123

FORMATTIME, Befehl [200](#)

RFC3339

FORMATTIME, Befehl [201](#)

ROLLBACK

CONNECT PROCESS [82](#)
 CONVERSE, Befehl (Nicht-z/OS Communications Server) [101](#)
 EXTRACT ATTRIBUTES (APPC) [176](#)
 EXTRACT ATTRIBUTES (MRO) [177](#)
 FREE (APPC) [203](#)
 FREE (MRO) [206](#)
 GDS ALLOCATE [212](#)
 GDS CONNECT PROCESS [216](#)
 GDS EXTRACT ATTRIBUTES [218](#)
 GDS FREE [220](#)
 GDS ISSUE ABEND [222](#)
 GDS ISSUE CONFIRMATION [223](#)
 GDS ISSUE ERROR [224](#)
 GDS ISSUE PREPARE [226](#)
 GDS ISSUE SIGNAL [227](#)
 GDS RECEIVE [229](#)
 GDS SEND [231](#)
 GDS WAIT [233](#)
 ISSUE ABEND, Befehl [287](#)
 ISSUE CONFIRMATION [293](#)
 ISSUE ERROR [304](#)
 ISSUE PREPARE [309](#)
 ISSUE SIGNAL (APPC) [319](#)
 RECEIVE (z/OS Communications Server), Befehl [423](#)
 RECEIVE, Befehl (MRO) [431](#)
 SEND (ohne z/OS Communications Server), Befehl [499](#)
 SEND (z/OS Communications Server), Befehl [494](#)
 WAIT CONVID, Befehl [630](#)

SEND

CONNECT PROCESS [82](#)
 CONVERSE, Befehl (Nicht-z/OS Communications Server) [101](#)
 EXTRACT ATTRIBUTES (APPC) [176](#)
 EXTRACT ATTRIBUTES (MRO) [178](#)
 FREE (APPC) [203](#)
 FREE (MRO) [206](#)
 GDS ALLOCATE [212](#)
 GDS CONNECT PROCESS [216](#)
 GDS EXTRACT ATTRIBUTES [218](#)
 GDS FREE [220](#)
 GDS ISSUE ABEND [222](#)
 GDS ISSUE CONFIRMATION [223](#)
 GDS ISSUE ERROR [224](#)
 GDS ISSUE PREPARE [226](#)
 GDS ISSUE SIGNAL [227](#)
 GDS RECEIVE [229](#)
 GDS SEND [231](#)
 GDS WAIT [233](#)

CVDA-Werte (Forts.)

SEND (Forts.)

ISSUE ABEND, Befehl [287](#)
 ISSUE CONFIRMATION [293](#)
 ISSUE ERROR [304](#)
 ISSUE PREPARE [309](#)
 ISSUE SIGNAL (APPC) [319](#)
 RECEIVE (z/OS Communications Server), Befehl [423](#)
 RECEIVE, Befehl (MRO) [431](#)
 SEND (ohne z/OS Communications Server), Befehl [499](#)
 SEND (z/OS Communications Server), Befehl [494](#)
 WAIT CONVID, Befehl [630](#)

SRVCONVERT

WEB RECEIVE (Server), Befehl [689](#)
 WEB SEND, Befehl (Server) [704](#)

SSL

EXTRACT TCPIP, Befehl [186](#)

SUBSPACE

ASSIGN, Befehl [46](#)

SUPPORTED

EXTRACT TCPIP, Befehl [186](#)

SYNCFREE

CONNECT PROCESS [82](#)
 CONVERSE, Befehl (Nicht-z/OS Communications Server) [101](#)
 EXTRACT ATTRIBUTES (APPC) [176](#)
 EXTRACT ATTRIBUTES (MRO) [178](#)
 FREE (APPC) [204](#)
 FREE (MRO) [206](#)
 GDS ALLOCATE [212](#)
 GDS CONNECT PROCESS [216](#)
 GDS EXTRACT ATTRIBUTES [218](#)
 GDS FREE [220](#)
 GDS ISSUE ABEND [222](#)
 GDS ISSUE CONFIRMATION [223](#)
 GDS ISSUE ERROR [224](#)
 GDS ISSUE PREPARE [226](#)
 GDS ISSUE SIGNAL [227](#)
 GDS RECEIVE [229](#)
 GDS SEND [231](#)
 GDS WAIT [233](#)
 ISSUE ABEND, Befehl [288](#)
 ISSUE CONFIRMATION [293](#)
 ISSUE ERROR [304](#)
 ISSUE PREPARE [309](#)
 ISSUE SIGNAL (APPC) [319](#)
 RECEIVE (z/OS Communications Server), Befehl [423](#)
 RECEIVE, Befehl (MRO) [431](#)
 SEND (ohne z/OS Communications Server), Befehl [499](#)
 SEND (z/OS Communications Server), Befehl [494](#)
 WAIT CONVID, Befehl [630](#)

SYNCRECEIVE

CONNECT PROCESS [82](#)
 CONVERSE, Befehl (Nicht-z/OS Communications Server) [101](#)
 EXTRACT ATTRIBUTES (APPC) [176](#)
 EXTRACT ATTRIBUTES (MRO) [178](#)
 FREE (APPC) [204](#)
 FREE (MRO) [206](#)
 GDS ALLOCATE [212](#)

CVDA-Werte (Forts.)

SYNCRECEIVE (Forts.)

GDS CONNECT PROCESS [216](#)
 GDS EXTRACT ATTRIBUTES [218](#)
 GDS FREE [220](#)
 GDS ISSUE ABEND [222](#)
 GDS ISSUE CONFIRMATION [223](#)
 GDS ISSUE ERROR [224](#)
 GDS ISSUE PREPARE [226](#)
 GDS ISSUE SIGNAL [227](#)
 GDS RECEIVE [229](#)
 GDS SEND [231](#)
 GDS WAIT [233](#)
 ISSUE ABEND, Befehl [288](#)
 ISSUE CONFIRMATION [293](#)
 ISSUE ERROR [304](#)
 ISSUE PREPARE [309](#)
 ISSUE SIGNAL (APPC) [319](#)
 RECEIVE (z/OS Communications Server), Befehl [423](#)
 RECEIVE, Befehl (MRO) [431](#)
 SEND (ohne z/OS Communications Server), Befehl [499](#)
 SEND (z/OS Communications Server), Befehl [494](#)
 WAIT CONVID, Befehl [630](#)

SYNCSEND

CONNECT PROCESS [82](#)
 CONVERSE, Befehl (Nicht-z/OS Communications Server) [101](#)
 EXTRACT ATTRIBUTES (APPC) [176](#)
 EXTRACT ATTRIBUTES (MRO) [178](#)
 FREE (APPC) [204](#)
 FREE (MRO) [206](#)
 GDS ALLOCATE [212](#)
 GDS CONNECT PROCESS [216](#)
 GDS EXTRACT ATTRIBUTES [218](#)
 GDS FREE [220](#)
 GDS ISSUE ABEND [222](#)
 GDS ISSUE CONFIRMATION [223](#)
 GDS ISSUE ERROR [224](#)
 GDS ISSUE PREPARE [226](#)
 GDS ISSUE SIGNAL [227](#)
 GDS RECEIVE [229](#)
 GDS SEND [231](#)
 GDS WAIT [233](#)
 ISSUE ABEND, Befehl [288](#)
 ISSUE CONFIRMATION [293](#)
 ISSUE ERROR [304](#)
 ISSUE PREPARE [309](#)
 ISSUE SIGNAL (APPC) [319](#)
 RECEIVE (z/OS Communications Server), Befehl [423](#)
 RECEIVE, Befehl (MRO) [431](#)
 SEND (ohne z/OS Communications Server), Befehl [499](#)
 SEND (z/OS Communications Server), Befehl [495](#)
 WAIT CONVID, Befehl [630](#)

TASK

DEQ [142](#)
 ENQ [166](#)

TRACE

WEB CONVERSE, Befehl [650](#)
 WEB SEND, Befehl (Client) [715](#)

UOW

CVDA-Werte (Forts.)

UOW (Forts.)

DEQ [142](#)

ENQ [166](#)

USEREXECKEY

ASSIGN, Befehl [46](#)

CWA, Option

ADDRESS, Befehl [33](#)

CWALENG, Option

ASSIGN, Befehl [48](#)

D

DATA, Operand

DFHMDI [862](#)

DFHMSD [871](#)

DATA, Option

FREEMAIN, Befehl [208](#)

FREEMAIN64, Befehl [210](#)

DATA1, Option

MONITOR, Befehl [340](#)

DATA2, Option

MONITOR, Befehl [341](#)

DATALength, Option

INQUIRE CONTAINER, Befehl [273](#)

LINK, Befehl [324](#)

DATAONLY, Option

DOCUMENT RETRIEVE, Befehl [151](#)

SEND MAP MAPPINGDEV, Befehl [514](#)

SEND MAP, Befehl [507](#)

DATAPointer, Option

FREEMAIN, Befehl [208](#)

FREEMAIN64, Befehl [210](#)

DATAStr, Option

BUILD ATTACH (LUTYPE6.1), Befehl [63](#)

BUILD ATTACH (MRO), Befehl [65](#)

EXTRACT ATTACH (LUTYPE6.1), Befehl [170](#)

EXTRACT ATTACH (MRO), Befehl [173](#)

DATAType, Option

PUT CONTAINER (CHANNEL), Befehl [357](#)

PUT64 CONTAINER, Befehl [361](#)

DATCONTAINER, Option

TRANSFORM XMLTODATA, Befehl [608](#)

DATE, Option

FORMATTIME, Befehl [199](#)

DATEFORM, Option

FORMATTIME, Befehl [199](#)

Dateien (Datasets)

abfragen [311](#)

Datensätze aktualisieren [313](#)

Datensätze hinzufügen [290](#)

Datensätze lesen [312](#)

Verarbeitung beenden [296](#)

Dateiname

Definition [4–6](#), [8](#), [9](#)

Dateiname, Argument, CICS-Befehlsformat [2](#)

Dateisteuerung

Befehle [21](#)

Browseoperation beenden [159](#)

Datensatz aktualisieren [467](#)

exklusive Steuerung freigeben [611](#)

nächsten Datensatz lesen [387](#)

neuen Datensatz schreiben [727](#)

Start für Suchvorgang angeben [583](#)

Dateisteuerung (Forts.)

vorherigen Datensatz lesen [398](#)

VSAM-Datensätze löschen [121](#)

Daten

an neue Tasks übergeben [565](#)

Daten an Ausgabeeinheit senden [316](#)

Daten löschen

benannter Zähler [134](#)

Dateisteuerungsdatensätze [121](#)

temporäre Speicherwarteschlangen [139](#)

Warteschlangen für transiente Daten [137](#)

Datenbereich, Argument

CICS-Befehlsformat [2](#)

Definition [1](#)

Datenbereich64

Befehlsformat [2](#)

Datensätze

aktualisieren [313](#), [467](#)

exklusive Steuerung freigeben [611](#)

lesen [312](#), [374](#)

nächste Nummer anfordern [305](#)

neu schreiben [727](#)

neue schreiben (hinzufügen) [290](#)

VSAM löschen [121](#)

Datentabellen

CICS/benutzerverwaltet/Coupling Facility

READ, Befehl [374](#)

CICS/benutzerverwaltet/Coupling-Facility

DELETE, Befehl [121](#)

ENDBR, Befehl [159](#)

READNEXT, Befehl [387](#)

READPREV, Befehl [398](#)

RESETBR, Befehl [449](#)

REWRITE, Befehl [467](#)

STARTBR, Befehl [583](#)

UNLOCK, Befehl [611](#)

WRITE, Befehl [727](#)

Datenübertragungssystem IBM 3770

logische Einheit mit Stapelverarbeitung [490](#)

Datenwert, Argument

CICS-Befehlsformat [2](#)

Definition [1](#)

DATESEP, Option

FORMATTIME, Befehl [199](#)

DATESTRING, Option

CONVERTTIME, Befehl [104](#)

FORMATTIME, Befehl [199](#)

DAYCOUNT, Option

FORMATTIME, Befehl [200](#)

DAYOFMONTH, Option

DEFINE TIMER, Befehl [117](#)

FORMATTIME, Befehl [200](#)

DAYOFWEEK, Option

FORMATTIME, Befehl [200](#)

DAYOFYEAR, Option

DEFINE TIMER, Befehl [117](#)

DAYS, Option

DEFINE TIMER, Befehl [117](#)

DAYSLEFT, Option

VERIFY PASSWORD, Befehl [620](#)

VERIFY PHRASE, Befehl [623](#)

DCOUNTER, Option

DELETE DCOUNTER, Befehl [134](#)

GET DCOUNTER, Befehl [240](#)

DCOUNTER, Option (*Forts.*)
 QUERY DCOUNT, Befehl [364](#)
 REWIND DCOUNT, Befehl [464](#)
 UPDATE DCOUNT, Befehl [615](#)

DDMMYY, Option
 FORMATIME, Befehl [200](#)

DDMMYYYY, Option
 FORMATIME, Befehl [200](#)

DEBKEY, Option
 READ, Befehl [377](#)
 STARTBR, Befehl [585](#)

DEBREC, Option
 READ, Befehl [377](#)
 STARTBR, Befehl [585](#)

DEFAULT, Option
 CONVERSE (Nicht-z/OS Communications Server), Befehl [100](#)
 CONVERSE (z/OS Communications Server), Befehl [94](#)
 SEND (ohne z/OS Communications Server), Befehl [498](#)
 SEND (z/OS Communications Server), Befehl [493](#)
 SEND CONTROL, Befehl [502](#)
 SEND MAP, Befehl [507](#)
 SEND TEXT NOEDIT, Befehl [529](#)
 SEND TEXT, Befehl [521](#)

DEFINE ACTIVITY, Befehl [105](#)

DEFINE COMPOSITE EVENT, Befehl [107](#)

DEFINE COUNTER, Befehl [109](#)

DEFINE DCOUNT, Befehl [109](#)

DEFINE INPUT EVENT, Befehl [112](#)

DEFINE PROCESS, Befehl [113](#)

DEFINE TIMER, Befehl [115](#)

DEFRESP, Option
 CONVERSE (Nicht-z/OS Communications Server), Befehl [100](#)
 CONVERSE (z/OS Communications Server), Befehl [94](#)
 ISSUE ADD, Befehl [291](#)
 ISSUE ERASE, Befehl [301](#)
 ISSUE REPLACE, Befehl [314](#)
 ISSUE SEND, Befehl [317](#)
 SEND (ohne z/OS Communications Server), Befehl [498](#)
 SEND (z/OS Communications Server), Befehl [493](#)

DEFSCRNHT, Option
 ASSIGN, Befehl [48](#)

DEFSCRNWD, Option
 ASSIGN, Befehl [48](#)

DELAY, Befehl [118](#)

DELETE ACTIVITY, Befehl [129](#)

DELETE CHANNEL, Befehl [130](#)

DELETE CONTAINER (BTS), Befehl [131](#)

DELETE CONTAINER (CHANNEL), Befehl [133](#)

DELETE COUNTER, Befehl [134](#)

DELETE DCOUNT, Befehl [134](#)

DELETE EVENT, Befehl [136](#)

DELETE TIMER, Befehl [137](#)

DELETE, Befehl [121](#)

DELETE, Option
 SPOOLCLOSE, Befehl [550](#)

DELETEQ TD, Befehl [137](#)

DELETEQ TS, Befehl [139](#)

DELIMITER, Option
 ASSIGN, Befehl [48](#)

DEQ, Befehl [140](#)

DESTCOUNT, Option
 ASSIGN, Befehl [48](#)

DESTID, Option
 ASSIGN, Befehl [48](#)
 ISSUE ABORT, Befehl [289](#)
 ISSUE ADD, Befehl [291](#)
 ISSUE END, Befehl [297](#)
 ISSUE ERASE, Befehl [301](#)
 ISSUE NOTE, Befehl [306](#)
 ISSUE QUERY, Befehl [311](#)
 ISSUE REPLACE, Befehl [314](#)
 ISSUE SEND, Befehl [317](#)
 ISSUE WAIT, Befehl [321](#)

DESTIDLENG, Option
 ASSIGN, Befehl [48](#)
 ISSUE ABORT, Befehl [289](#)
 ISSUE ADD, Befehl [291](#)
 ISSUE END, Befehl [297](#)
 ISSUE ERASE, Befehl [301](#)
 ISSUE NOTE, Befehl [306](#)
 ISSUE QUERY, Befehl [311](#)
 ISSUE REPLACE, Befehl [314](#)
 ISSUE SEND, Befehl [317](#)
 ISSUE WAIT, Befehl [321](#)

DET, Wert
 DFHMDF [851](#)
 DFH2980-Struktur [427](#)
 DFHAID, AID-Zeichenliste [846](#)
 DFHBMSCA, Standardliste für Attribute und Druckersteuerzeichen, BMS [841](#)
 DFHEAI, Schnittstellenprozessor [13](#)
 DFHECALL, Makro [13](#)
 DFHEIBLK, Copybook [13](#)
 DFHEIEND, Makro [13](#)
 DFHEIENT, Makro
 Beschreibung [13](#)
 DFHEIGBL, Makro [13](#)
 DFHEIRET, Makro [13](#)
 DFHEISTG, Makro [13](#)
 DFHMDF, Makro [851](#)
 DFHMDI, Makro [862](#)
 DFHMSD, Makro [871](#)
 DFHMSRCA, MSR-Steuerwertkonstanten [845](#)
 DFHPDI, Makro [881](#)
 DFHPSD, Makro [882](#)
 DFHRESP, integrierte Funktion [10](#)
 Diagnoseservices, Befehle [20](#)
 Dialog mit Terminal oder logischer Einheit [833](#)

DIGESTTYPE, Option
 BIF DIGEST, Befehl [61](#)

DISABLED, Bedingung
 DELETE, Befehl [121](#)
 DELETEQ TD, Befehl [138](#)
 READ, Befehl [381](#)
 READQ TD, Befehl [409](#)
 STARTBR, Befehl [587](#)
 UNLOCK, Befehl [613](#)
 WRITE, Befehl [730](#)
 WRITEQ TD, Befehl [741](#)

DOCSIZE, Option
 DOCUMENT INSERT, Befehl [148](#)

DOCSTATUS, Option
 WEB CONVERSE, Befehl [649](#)
 WEB SEND, Befehl (Client) [714](#)
 WEB SEND, Befehl (Server) [703](#)

DOCTOKEN, Option

- DOCTOKEN, Option (*Forts.*)
 - DOCUMENT RETRIEVE, Befehl [151](#)
 - DOCUMENT SET, Befehl [153](#)
 - WEB CONVERSE, Befehl [649](#)
 - WEB RETRIEVE, Befehl [699](#)
 - WEB SEND, Befehl (Client) [714](#)
 - WEB SEND, Befehl (Server) [703](#)
- DOCUMENT CREATE, Befehl [142](#)
- DOCUMENT DELETE, Befehl [146](#)
- DOCUMENT INSERT, Befehl [146](#)
- DOCUMENT RETRIEVE, Befehl [150](#)
- DOCUMENT SET, Befehl [152](#)
- DOCUMENT, Option
 - DOCUMENT INSERT, Befehl [148](#)
- Dokument
 - erstellen [142](#)
 - löschen [146](#)
 - Symbole der Symboltabelle hinzufügen [152](#)
- Dokumentservices
 - Befehle [20](#)
- DORMANT, Modus einer Aktivität [271](#)
- DRK, Wert
 - DFHMDF [851](#)
- Drucken von angezeigten Informationen [836](#)
- Druckersteuerzeichenliste, DFHBMSCA [841](#)
- DS3270, Option
 - ASSIGN, Befehl [49](#)
- DSATTS, Operand
 - DFHMDI [862](#)
 - DFHMSD [871](#)
- DSECT, Operand
 - DFHMSD [871](#)
- DSECT, Wert
 - DFHMSD [871](#)
- DSSCS, Option
 - ASSIGN, Befehl [48](#)
- DSSTAT, Bedingung
 - ISSUE RECEIVE, Befehl [313](#)
- DUMP TRANSACTION, Befehl [154](#)
- DUMPCODE, Option
 - DUMP TRANSACTION, Befehl [155](#)
- DUMPID, Option
 - DUMP TRANSACTION, Befehl [155](#)
- DUPKEY, Bedingung
 - DELETE, Befehl [121](#)
 - READ, Befehl [382](#)
 - READNEXT, Befehl [394](#)
 - READPREV, Befehl [404](#)
- DUPREC, Bedingung
 - REWRITE, Befehl [469](#)
 - WRITE, Befehl [730](#)
- Dynamische Zuordnung [555](#)

E

- ECADDR, Option
 - WAIT EVENT, Befehl [631](#)
- ECBLIST, Option
 - WAIT EXTERNAL, Befehl [633](#)
 - WAITCICS, Befehl [640](#)
- EDF, Execution Diagnostic Facility [565](#), [579](#)
- EIB, Option
 - ADDRESS, Befehl [33](#)
- EIBAID

- EIBAID (*Forts.*)
 - Feldinhalt untersuchen [846](#)
- Eingabeoperation ohne Daten [837](#)
- Einstieg in Assemblerprogramm [13](#)
- Einzelthread, bei JES verwendeter [552](#)
- ELEMNAME, Option
 - TRANSFORM XMLTODATA, Befehl [609](#)
- ELEMNAMELEN, Option
 - TRANSFORM XMLTODATA, Befehl [609](#)
- ELEMNS, Option
 - TRANSFORM XMLTODATA, Befehl [609](#)
- ELEMNSLEN, Option
 - TRANSFORM XMLTODATA, Befehl [609](#)
- END, Bedingung
 - GETNEXT ACTIVITY, Befehl [253](#)
 - GETNEXT CONTAINER (CHANNEL), Befehl [255](#)
 - GETNEXT CONTAINER, Befehl [254](#)
 - GETNEXT EVENT, Befehl [256](#)
 - GETNEXT PROCESS, Befehl [257](#)
 - RETRIEVE REATTACH EVENT, Befehl [459](#)
 - RETRIEVE SUBEVENT, Befehl [460](#)
- ENDACTIVITY, Option
 - RETURN, Befehl [462](#)
- ENDBR, Befehl [159](#)
- ENDBROWSE ACTIVITY, Befehl [161](#)
- ENDBROWSE CONTAINER (CHANNEL), Befehl [162](#)
- ENDBROWSE CONTAINER, Befehl [162](#)
- ENDBROWSE EVENT, Befehl [163](#)
- ENDBROWSE PROCESS, Befehl [163](#)
- ENDBROWSE TIMER, Befehl [164](#)
- ENDDATA, Bedingung
 - RETRIEVE, Befehl [456](#)
- ENDFILE, Bedingung
 - READNEXT, Befehl [394](#)
 - READPREV, Befehl [404](#)
 - SPOOLREAD, Befehl [560](#)
 - WEB READNEXT FORMFIELD, Befehl [681](#)
 - WEB READNEXT HTTPHEADER, Befehl [682](#)
 - WEB READNEXT QUERYPARM, Befehl [684](#)
- ENDFILE, Option
 - ISSUE ENDOUTPUT, Befehl [299](#)
- ENDINPT, Bedingung
 - RECEIVE, Befehl (ohne z/OS Communications Server) [431](#)
- ENDOUTPUT, Option
 - ISSUE ENDFILE, Befehl [298](#)
- Englische und Katakana-Zeichen gemischt [99](#), [433](#), [438](#)
- ENQ, Befehl [164](#)
- ENQBUSY, Bedingung
 - ENQ, Befehl [167](#)
- ENTER TRACEID, Befehl
 - Überwachungsaspekte, durch Befehl MONITOR ersetzt [340](#)
- ENTER TRACENUM, Befehl [167](#)
- ENTER, Option
 - HANDLE AID, Befehl [265](#)
- Entfernen aller ungeschützten Felder [837](#)
- ENTRY, Option
 - LOAD, Befehl [338](#)
- ENTRYNAME, Option
 - MONITOR, Befehl [341](#)
- ENVDEFERR, Bedingung
 - RETRIEVE, Befehl [457](#)
- EOC, Bedingung

EOC, Bedingung (*Forts.*)
 ALLOCATE (LUTYPE6.1), Befehl [39](#)
 CONVERSE (Nicht-z/OS Communications Server), Befehl [102](#)
 CONVERSE (z/OS Communications Server), Befehl [96](#)
 ISSUE RECEIVE, Befehl [313](#)
 RECEIVE (z/OS Communications Server), Befehl [423](#)
 RECEIVE MAP, Befehl [434](#)
 RECEIVE PARTN, Befehl [439](#)
 RECEIVE, Befehl (ohne z/OS Communications Server) [431](#)
 WAIT TERMINAL, Befehl [639](#)

EOS, Bedingung
 CONVERSE (z/OS Communications Server), Befehl [96](#)
 ISSUE RECEIVE, Befehl [313](#)
 RECEIVE (z/OS Communications Server), Befehl [424](#)
 RECEIVE MAP, Befehl [434](#)
 RECEIVE PARTN, Befehl [439](#)

EOF, Bedingung
 CONVERSE (Nicht-z/OS Communications Server), Befehl [102](#)
 RECEIVE, Befehl (ohne z/OS Communications Server) [431](#)

EQUAL, Option
 READ, Befehl [377](#)
 RESETBR, Befehl [450](#)
 STARTBR, Befehl [585](#)

ERASE, Option
 CONVERSE (Nicht-z/OS Communications Server), Befehl [100](#)
 CONVERSE (z/OS Communications Server), Befehl [94](#)
 SEND (ohne z/OS Communications Server), Befehl [498](#)
 SEND (z/OS Communications Server), Befehl [493](#)
 SEND CONTROL, Befehl [502](#)
 SEND MAP MAPPINGDEV, Befehl [514](#)
 SEND MAP, Befehl [507](#)
 SEND TEXT NOEDIT, Befehl [529](#)
 SEND TEXT, Befehl [521](#)

ERASEAUP, Option
 SEND CONTROL, Befehl [502](#)
 SEND MAP MAPPINGDEV, Befehl [514](#)
 SEND MAP, Befehl [508](#)

Ereignisbezogene Befehle
 CHECK TIMER [80](#)
 DEFINE COMPOSITE EVENT [107](#)
 DEFINE INPUT EVENT [112](#)
 DEFINE TIMER [115](#)
 DELETE EVENT [136](#)
 DELETE TIMER [137](#)
 ENDBROWSE EVENT [163](#)
 ENDBROWSE TIMER [164](#)
 FORCE TIMER [196](#)
 GETNEXT EVENT [255](#)
 GETNEXT TIMER [258](#)
 INQUIRE EVENT [274](#)
 INQUIRE TIMER [277](#)
 REMOVE SUBEVENT [441](#)
 RETRIEVE REATTACH EVENT [458](#)
 RETRIEVE SUBEVENT [459](#)
 STARTBROWSE EVENT [594](#)
 STARTBROWSE TIMER [596](#)
 TEST EVENT [601](#)

Ereignisse, Zeitgeber
 Steuerbereich, Zeitgeber [349](#)

Ereignisse, Zeitgeber (*Forts.*)
 Überwachungspunkt [340](#)
 warten auf [630](#)

Ereignisverarbeitung, Befehle [21](#)

ERRORMSG, Option
 ASSIGN, Befehl [49](#)

ERRORMSGELN, Option
 ASSIGN, Befehl [49](#)

ERRTERM, Option
 ROUTE, Befehl [474](#)

Erstellen eines Journalsatzes [322](#), [734](#)

Erweiterte relative Byteadresse (XRBA) [121](#)

ESDS (Datei in Zugangsfolge)
 DELETE, Befehl [121](#)
 READ [379](#), [381](#)

ESDS (Entry-Sequenced Data Set, Datei in Zugangsfolge)
 READNEXT [394](#)
 READPREV [404](#)
 RESETBR [451](#)
 STARTBR [587](#)
 STARTBR, Befehl [586](#)
 WRITE, Befehl [728](#)

ESM
 ACEE-Zeiger [33](#)
 QUERY SECURITY, Befehl, NOTFND, Bedingung [373](#)
 QUERY SECURITY, Befehl, RESCLASS, Option [368](#)
 USERNAME [57](#)

ESM, externer Sicherheitsmanager [565](#), [579](#)

ESMAPNAME, Option
 REQUEST PASSTICKET [443](#), [445](#)

ESMREASON, Option
 CHANGE PASSWORD, Befehl [74](#)
 CHANGE PHRASE, Befehl [72](#)
 SIGNON TOKEN, Befehl [540](#)
 SIGNON, Befehl [535](#)
 VERIFY PASSWORD, Befehl [620](#)
 VERIFY PHRASE, Befehl [623](#)
 VERIFY TOKEN, Befehl [627](#)

ESMRESP, Option
 CHANGE PASSWORD, Befehl [74](#)
 CHANGE PHRASE, Befehl [72](#)
 REQUEST PASSTICKET [443](#), [445](#)
 SIGNON TOKEN, Befehl [540](#)
 SIGNON, Befehl [535](#)
 VERIFY PASSWORD, Befehl [620](#)
 VERIFY PHRASE, Befehl [623](#)
 VERIFY TOKEN, Befehl [627](#)

EVENT, Option
 ADD SUBEVENT, Befehl [31](#)
 DEFINE ACTIVITY, Befehl [105](#)
 DEFINE COMPOSITE EVENT, Befehl [108](#)
 DEFINE INPUT EVENT, Befehl [113](#)
 DEFINE TIMER, Befehl [117](#)
 DELETE EVENT, Befehl [136](#)
 GETNEXT EVENT, Befehl [256](#)
 INQUIRE ACTIVITYID, Befehl [271](#)
 INQUIRE EVENT, Befehl [274](#)
 INQUIRE TIMER, Befehl [277](#)
 REMOVE SUBEVENT, Befehl [441](#)
 RETRIEVE REATTACH EVENT, Befehl [459](#)
 RETRIEVE SUBEVENT, Befehl [460](#)
 SIGNAL EVENT, Befehl [532](#)
 TEST EVENT, Befehl [601](#)

EVENTERR, Bedingung

- EVENTERR, Bedingung (*Forts.*)
 - ADD SUBEVENT, Befehl [31](#)
 - DEFINE ACTIVITY, Befehl [106](#)
 - DEFINE COMPOSITE EVENT, Befehl [108](#)
 - DEFINE INPUT EVENT, Befehl [113](#)
 - DEFINE TIMER, Befehl [117](#)
 - DELETE EVENT, Befehl [136](#)
 - INQUIRE EVENT, Befehl [275](#)
 - LINK ACQPROCESS, Befehl [332](#)
 - LINK ACTIVITY, Befehl [335](#)
 - REMOVE SUBEVENT, Befehl [441](#)
 - RETRIEVE SUBEVENT, Befehl [460](#)
 - RUN, Befehl [479](#)
 - SIGNAL EVENT, Befehl [532](#)
 - TEST EVENT, Befehl [601](#)
- EVENTTYPE, Option
 - GETNEXT EVENT, Befehl [256](#)
 - INQUIRE EVENT, Befehl [275](#)
 - RETRIEVE REATTACH EVENT, Befehl [459](#)
 - RETRIEVE SUBEVENT, Befehl [460](#)
- EWASUPP, Option
 - ASSIGN, Befehl [49](#)
- EXACTMATCH, Option
 - INVOKE APPLICATION, Befehl [279](#)
- EXCEPTION, Option
 - ENTER TRACENUM, Befehl [168](#)
- EXEC CICS, Befehlsformat [1](#)
- Execution Diagnostic Facility (EDF) [565](#), [579](#)
- Exit aus ASM-Programm [13](#)
- Exit, Wiederherstellung nach abnormaler Beendigung [263](#)
- EXPIRED, Bedingung
 - DELAY, Befehl [118](#)
 - POST, Befehl [351](#)
 - WRITE OPERATOR, Befehl [740](#)
- EXPIRYTIME, Option
 - VERIFY PASSWORD, Befehl [620](#)
 - VERIFY PHRASE, Befehl [623](#)
- EXTATT, Operand
 - DFHMDI [862](#)
 - DFHMSD [871](#)
- EXTDS, Option
 - ASSIGN, Befehl [49](#)
- Externer Sicherheitsmanager (ESM) [565](#), [579](#)
- EXTRACT ATTACH (LUTYPE6.1), Befehl [169](#)
- EXTRACT ATTACH (MRO), Befehl [173](#)
- EXTRACT ATTRIBUTES (APPC), Befehl [176](#)
- EXTRACT ATTRIBUTES (MRO), Befehl [177](#)
- EXTRACT CERTIFICATE, Befehl [178](#)
- EXTRACT LOGONMSG, Befehl [181](#)
- EXTRACT PROCESS, Befehl [182](#)
- EXTRACT TCPIP, Befehl [183](#)
- EXTRACT TCT, Befehl [187](#)
- EXTRACT WEB, Befehl [188](#)

F

- FACILITY, Option
 - ASSIGN, Befehl [49](#)
- FACILITYTOKEN, Option
 - RUN, Befehl [479](#)
- FCI, Option
 - ASSIGN, Befehl [49](#), [60](#)
- FCT, Option
 - DUMP TRANSACTION, Befehl [155](#)

- Feld
 - Informationen extrahieren [674](#)
- Felddefinitions makro, BMS [849](#)
- Feldtrennzeichen, Operand [862](#), [871](#)
- Fernschreiber
 - Nachrichten [835](#)
 - Programmierung [835](#)
- FIELD, Option
 - BIF DEEDIT, Befehl [60](#)
- FIELD, Wert
 - DFHMDI [862](#)
 - DFHMSD [871](#)
- FIELDS, Operand
 - DFHMDI [862](#)
- FILE, Option
 - DELETE, Befehl [121](#)
 - ENDBR, Befehl [160](#)
 - READ, Befehl [377](#)
 - READNEXT, Befehl [390](#)
 - READPREV, Befehl [401](#)
 - RESETBR, Befehl [450](#)
 - REWRITE, Befehl [468](#)
 - STARTBR, Befehl [585](#)
 - UNLOCK, Befehl [612](#)
 - WRITE, Befehl [728](#)
- FILENOTFOUND, Bedingung
 - DELETE, Befehl [121](#)
 - ENDBR, Befehl [160](#)
 - READ, Befehl [382](#)
 - READNEXT, Befehl [394](#)
 - READPREV, Befehl [405](#)
 - RESETBR, Befehl [451](#)
 - REWRITE, Befehl [469](#)
 - STARTBR, Befehl [587](#)
 - UNLOCK, Befehl [613](#)
 - WRITE, Befehl [730](#)
- FIRESTATUS, Option
 - GETNEXT EVENT, Befehl [256](#)
 - INQUIRE EVENT, Befehl [275](#)
 - TEST EVENT, Befehl [601](#)
- FIRST, Wert
 - DFHMDI [862](#)
- FLDSEP, Operand
 - DFHMDI [862](#)
 - DFHMSD [871](#)
- FLENGTH, Option
 - DUMP TRANSACTION, Befehl [155](#)
 - GDS RECEIVE, Befehl [228](#)
 - GDS SEND, Befehl [230](#)
 - GET CONTAINER (BTS), Befehl [234](#)
 - GET CONTAINER (CHANNEL), Befehl [237](#)
 - GET64 CONTAINER, Befehl [260](#)
 - GETMAIN, Befehl [247](#)
 - GETMAIN64, Befehl [250](#)
 - LOAD, Befehl [338](#)
 - PUT CONTAINER (BTS), Befehl [354](#)
 - PUT CONTAINER (CHANNEL), Befehl [357](#)
 - PUT64 CONTAINER, Befehl [361](#)
 - RECEIVE (z/OS Communications Server), Befehl [422](#)
 - RECEIVE, Befehl (ohne z/OS Communications Server) [429](#)
 - SEND (ohne z/OS Communications Server), Befehl [498](#)
 - SEND (z/OS Communications Server), Befehl [493](#)
 - SIGNAL EVENT, Befehl [532](#)

FLENGTH, Option (*Forts.*)
 SPOOLWRITE, Befehl [563](#)
 Vollwortalternative für LENGTH [833](#)
 WRITE JOURNALNAME, Befehl [735](#)
 FMH, Option
 CONVERSE (Nicht-z/OS Communications Server), Befehl [100](#)
 CONVERSE (z/OS Communications Server), Befehl [94](#)
 SEND (ohne z/OS Communications Server), Befehl [498](#)
 SEND (z/OS Communications Server), Befehl [494](#)
 START, Befehl [570](#)
 FMHPARM, Option
 SEND MAP, Befehl [508](#)
 SEND PAGE, Befehl [517](#)
 SEND TEXT, Befehl [521](#)
 FOLD. Operand
 DFHMSD [871](#)
 FOR, Option
 DELAY, Befehl [118](#)
 FORCE TIMER, Befehl [196](#)
 FORMATTIME, Befehl [197](#)
 FORMFEED, Option
 SEND CONTROL, Befehl [502](#)
 SEND MAP MAPPINGDEV, Befehl [514](#)
 SEND MAP, Befehl [508](#)
 SEND TEXT, Befehl [521](#)
 FORMFIELD, Option
 WEB READ FORMFIELD, Befehl [675](#)
 WEB READNEXT FORMFIELD, Befehl [680](#)
 WEB STARTBROWSE FORMFIELD, Befehl [720](#)
 Formularfeld
 Informationen extrahieren [674](#)
 FREE (APPC), Befehl [203](#)
 FREE (LUTYPE6.1), Befehl [204](#)
 FREE (MRO), Befehl [205](#)
 FREE, Befehl [202](#)
 FREEKB, Option
 SEND CONTROL, Befehl [502](#)
 SEND MAP MAPPINGDEV, Befehl [514](#)
 SEND MAP, Befehl [508](#)
 SEND TEXT NOEDIT, Befehl [529](#)
 SEND TEXT, Befehl [522](#)
 FREEKB, Wert
 DFHMDI [862](#)
 DFHMSD [871](#)
 FREEMAIN, Befehl [207](#)
 FREEMAIN64, Befehl [209](#)
 Freigeben der exklusiven Steuerung mit Befehl UNLOCK [611](#)
 FROM, Option
 CONVERSE (Nicht-z/OS Communications Server), Befehl [100](#)
 CONVERSE (z/OS Communications Server), Befehl [94](#)
 Vollwortalternative für FROMLENGTH [833](#)
 FROMLENGTH, Option
 CONVERSE (Nicht-z/OS Communications Server), Befehl [100](#)
 CONVERSE (z/OS Communications Server), Befehl [94](#)
 ENTER TRACENUM, Befehl [168](#)
 Vollwortalternative für Länge (FROMLENGTH) [833](#)
 WEB CONVERSE, Befehl [649](#)
 WEB SEND, Befehl (Client) [714](#)
 WEB SEND, Befehl (Server) [704](#)
 FROMPROCESS, Option
 MOVE CONTAINER (BTS), Befehl [344](#)
 FRSET, Option
 SEND CONTROL, Befehl [502](#)
 SEND MAP MAPPINGDEV, Befehl [514](#)
 SEND MAP, Befehl [508](#)
 FRSET, Wert
 DFHMDI [862](#)
 DFHMSD [871](#)
 FSET, Wert
 DFHMDI [862](#)
 FULLDATE, Option
 FORMATTIME [200](#)
 FUNCERR, Bedingung
 ISSUE ABORT, Befehl [290](#)
 ISSUE ADD, Befehl [292](#)
 ISSUE END, Befehl [298](#)
 ISSUE ERASE, Befehl [301](#)
 ISSUE NOTE, Befehl [306](#)

FUNCERR, Bedingung (*Forts.*)

ISSUE QUERY, Befehl [311](#)

ISSUE REPLACE, Befehl [315](#)

ISSUE SEND, Befehl [317](#)

ISSUE WAIT, Befehl [321](#)

G

GCHARS, Option

ASSIGN, Befehl [49](#)

GCODES, Option

ASSIGN, Befehl [50](#)

GDS (generalisierter Datenstrom) [17](#)

GDS ALLOCATE, Befehl [211](#)

GDS ASSIGN, Befehl [213](#)

GDS CONNECT PROCESS, Befehl [214](#)

GDS EXTRACT ATTRIBUTES, Befehl [217](#)

GDS EXTRACT PROCESS, Befehl [218](#)

GDS FREE, Befehl [219](#)

GDS ISSUE ABEND, Befehl [221](#)

GDS ISSUE CONFIRMATION, Befehl [222](#)

GDS ISSUE ERROR, Befehl [223](#)

GDS ISSUE PREPARE, Befehl [225](#)

GDS ISSUE SIGNAL, Befehl [226](#)

GDS RECEIVE, Befehl [227](#)

GDS SEND, Befehl [229](#)

GDS WAIT, Befehl [232](#)

Generalisierter Datenstrom (GDS) [17](#)

GENERIC, Option

DELETE, Befehl [121](#)

READ, Befehl [377](#)

RESETBR, Befehl [450](#)

STARTBR, Befehl [585](#)

Generische Anwendungs-ID (APPLID), XRF [45](#)

GET CONTAINER (BTS), Befehl [233](#)

GET CONTAINER (CHANNEL), Befehl [236](#)

GET COUNTER, Befehl [240](#)

GET DOUNTER, Befehl [240](#)

GETMAIN, Befehl [245](#)

GETMAIN64, Befehl [248](#)

GETNEXT ACTIVITY, Befehl [252](#)

GETNEXT CONTAINER (CHANNEL), Befehl [254](#)

GETNEXT CONTAINER, Befehl [253](#)

GETNEXT EVENT, Befehl [255](#)

GETNEXT PROCESS, Befehl [257](#)

GETNEXT TIMER, Befehl [258](#)

GINIT, Operand

DFHMDf [851](#)

Gleichgesetzte Symbole [7](#)

GMMI, Option

ASSIGN, Befehl [50](#)

Große COMMAREAs (Kanäle)

ASSIGN, Befehl [47](#)

DELETE CHANNEL, Befehl [130](#)

DELETE CONTAINER (CHANNEL), Befehl [133](#)

QUERY CHANNEL, Befehl [363](#)

Große COMMAREAs, Kanäle [130](#), [133](#), [236](#), [345](#), [355](#), [363](#), [460](#), [579](#), [759](#)

GROUPID, Option

SIGNON TOKEN, Befehl [540](#)

SIGNON, Befehl [535](#)

GRPNAME, Operand

DFHMDf [851](#)

GTEQ, Option

GTEQ, Option (*Forts.*)

READ, Befehl [378](#)

RESETBR, Befehl [450](#)

STARTBR, Befehl [585](#)

H

HANDLE ABEND, Befehl [263](#)

HANDLE AID, Befehl [265](#)

Hauptspeicher [245](#)

Hauptspeicher (64-Bit) [248](#)

Hauptspeicher freigeben [207](#), [209](#)

Hauptspeicher initialisieren [245](#)

Header

Browsing [658](#), [720](#)

nächsten abrufen [680](#)

HEADER, Operand

DFHMDI [862](#)

HEADER, Option

SEND TEXT, Befehl [522](#)

hmmss, Argument, CICS-Befehlsformat [2](#)

HIGHLIGHT, Operand

DFHMDf [851](#)

DFHMDI [862](#)

DFHMSD [871](#)

HIGHLIGHT, Option

ASSIGN, Befehl [50](#)

Hohe COMMAREA-Werte, Kanäle [130](#), [133](#), [236](#), [345](#), [355](#), [363](#), [460](#), [579](#), [759](#)

HOLD, Option

LOAD, Befehl [338](#)

HONEOM, Option

SEND CONTROL, Befehl [503](#)

SEND MAP, Befehl [508](#)

SEND TEXT NOEDIT, Befehl [529](#)

SEND TEXT, Befehl [522](#)

HOST, Option

WEB EXTRACT oder EXTRACT WEB, Befehl [189](#), [662](#)

WEB OPEN, Befehl [668](#)

WEB PARSE URL, Befehl [672](#)

Hostbefehlsprozessor, logische Einheit mit 3650- bzw. 3680- [489](#)

HOSTCODEPAGE, Option

Befehl WEB READ FORMFIELD [675](#)

WEB READ QUERYPARM, Befehl [679](#)

WEB RECEIVE, Befehl (Server) [687](#)

WEB SEND, Befehl (Server) [704](#)

WEB STARTBROWSE FORMFIELD, Befehl [720](#)

WEB STARTBROWSE QUERYPARM, Befehl [723](#)

Hostdialog, logische Einheit mit 3650-

(3270) [488](#)

(3653) [489](#)

Hostdialog, LU 3650

(3270) [89](#)

(3653) [90](#)

HOSTLENGTH, Option

WEB EXTRACT oder EXTRACT WEB, Befehl [190](#), [662](#)

WEB OPEN, Befehl [668](#)

WEB PARSE URL, Befehl [672](#)

HOSTTYPE, Option

WEB EXTRACT oder EXTRACT WEB, Befehl [190](#), [662](#)

WEB PARSE URL, Befehl [673](#)

HOURS, Option

DEFINE TIMER, Befehl [117](#)

HOURS, Option (*Forts.*)

DELAY, Befehl [118](#)

POST, Befehl [350](#)

ROUTE, Befehl [474](#)

START, Befehl [570](#)

HTAB, Operand

DFHMSD [871](#)

HTTPHEADER, Option

WEB READ HTTPHEADER, Befehl [677](#)

WEB READNEXT HTTPHEADER, Befehl [682](#)

WEB WRITE HTTPHEADER, Befehl [726](#)

HTTPMETHOD, Option

WEB EXTRACT oder EXTRACT WEB, Befehl [190](#), [662](#)

HTTPRNUM, Option

WEB OPEN, Befehl [668](#)

HTTPVERSION, Option

WEB EXTRACT oder EXTRACT WEB, Befehl [190](#), [662](#)

HTTPVNUM, Option

WEB OPEN, Befehl [668](#)

I

IC, Wert

DFHMDF [851](#)

IGNORE CONDITION, Befehl [269](#)

IGREQCD, Bedingung

CONVERSE (z/OS Communications Server), Befehl [97](#)

ISSUE SEND, Befehl [317](#)

SEND (z/OS Communications Server), Befehl [495](#)

SEND CONTROL, Befehl [504](#)

SEND MAP, Befehl [511](#)

SEND PAGE, Befehl [518](#)

SEND TEXT MAPPED, Befehl [527](#)

SEND TEXT NOEDIT, Befehl [530](#)

SEND TEXT, Befehl [525](#)

IGREQID, Bedingung

ROUTE, Befehl [476](#)

SEND CONTROL, Befehl [504](#)

SEND MAP, Befehl [511](#)

SEND TEXT MAPPED, Befehl [527](#)

SEND TEXT NOEDIT, Befehl [530](#)

SEND TEXT, Befehl [525](#)

ILLOGIC, Bedingung

DELETE, Befehl [121](#)

ENDBR, Befehl [160](#)

ENDBROWSE ACTIVITY, Befehl [161](#)

ENDBROWSE CONTAINER (CHANNEL), Befehl [163](#)

ENDBROWSE CONTAINER, Befehl [162](#)

ENDBROWSE PROCESS, Befehl [164](#)

GETNEXT ACTIVITY, Befehl [253](#)

GETNEXT CONTAINER (CHANNEL), Befehl [255](#)

GETNEXT CONTAINER, Befehl [254](#)

GETNEXT PROCESS, Befehl [257](#)

INQUIRE PROCESS, Befehl [276](#)

READ, Befehl [382](#)

READNEXT, Befehl [394](#)

READPREV, Befehl [405](#)

RESETBR, Befehl [451](#)

REWRITE, Befehl [469](#)

SPOOL OPEN INPUT, Befehl [553](#)

SPOOL OPEN OUTPUT, Befehl [558](#)

SPOOL READ, Befehl [561](#)

STARTBR, Befehl [587](#)

UNLOCK, Befehl [613](#)

ILLOGIC, Bedingung (*Forts.*)

WEB STARTBROWSE FORMFIELD, Befehl [721](#)

WEB STARTBROWSE HTTPHEADER, Befehl [722](#)

WEB STARTBROWSE QUERYPARM, Befehl [723](#)

WRITE, Befehl [730](#)

IMMEDIATE, Option

RETURN, Befehl [462](#)

Impliziter Befehl SPOOLCLOSE [552](#)

INBFMH, Bedingung

CONVERSE (Nicht-z/OS Communications Server), Befehl [102](#)

CONVERSE (z/OS Communications Server), Befehl [97](#)

RECEIVE (z/OS Communications Server), Befehl [424](#)

RECEIVE, Befehl (ohne z/OS Communications Server) [431](#)

INCREMENT, Option

Befehl REWIND DOUNTER [464](#)

GET COUNTER, Befehl [240](#)

GET DOUNTER, Befehl [240](#)

REWIND COUNTER, Befehl [464](#)

INITIAL, Modus einer Aktivität [271](#)

INITIAL, Operand

DFHMDF [851](#)

INITIMG, Option

GETMAIN, Befehl [247](#)

INITPARM, Option

ASSIGN, Befehl [50](#)

INITPARMLEN, Option

ASSIGN, Befehl [50](#)

INPARTN, Option

ASSIGN, Befehl [50](#)

RECEIVE MAP, Befehl [433](#)

INPUTEVENT, Option

LINK ACQPROCESS, Befehl [332](#)

LINK ACTIVITY, Befehl [335](#)

RUN, Befehl [479](#)

INPUTMSG, Option

LINK, Befehl [325](#)

RETURN, Befehl [463](#)

XCTL, Befehl [761](#)

INPUTMSGLEN, Option

ASSIGN, Befehl [50](#)

LINK, Befehl [325](#)

RETURN, Befehl [463](#)

XCTL, Befehl [761](#)

INQUIRE ACTIVITYID, Befehl [270](#)

INQUIRE CONTAINER, Befehl [272](#)

INQUIRE EVENT, Befehl [274](#)

INQUIRE PROCESS, Befehl [276](#)

INQUIRE TIMER, Befehl [277](#)

Integrierte Funktionen

Befehle [19](#)

Interaktive logische 3770-Einheit

RECEIVE, Befehl [420](#)

SEND, Befehl [490](#)

Interaktive logische Einheiten [91](#), [420](#), [490](#)

Interpreter, logische Einheit [3650](#)

Befehl SEND (z/OS Communications Server) [488](#)

CONVERSE, Befehl [89](#)

ISSUE EODS, Befehl [299](#)

ISSUE LOAD, Befehl [304](#)

RECEIVE, Befehl [419](#)

INTERVAL, Option

DELAY, Befehl [118](#)

INTERVAL, Option (*Forts.*)
 POST, Befehl [350](#)
 ROUTE, Befehl [474](#)
 START, Befehl [570](#)
 Intervallsteuerung
 Abrufen von gespeicherten Daten für eine Task [455](#)
 aktuelle Uhrzeit anfordern [41](#)
 ASKTIME-Optionen [41](#)
 auf Auftreten eines Ereignisses warten [630](#)
 Befehle [21](#)
 Benachrichtigung nach Ablauf einer angegebenen Zeit [348](#)
 CANCEL, Optionen [68](#)
 DELAY-Optionen [118](#)
 FORMATTIME, Optionen [199](#)
 Intervallsteuerung abbrechen, Befehl [67](#)
 Task starten [565](#)
 Verarbeitung von Task verzögern [118](#)
 INTO, Option
 CONVERSE (Nicht-z/OS Communications Server), Befehl [100](#)
 CONVERSE (z/OS Communications Server), Befehl [94](#)
 DOCUMENT RETRIEVE, Befehl [151](#)
 EXTRACT LOGONMSG, Befehl [181](#)
 GDS RECEIVE, Befehl [228](#)
 GET CONTAINER (BTS), Befehl [234](#)
 GET CONTAINER (CHANNEL), Befehl [237](#)
 GET64 CONTAINER, Befehl [261](#)
 ISSUE RECEIVE, Befehl [312](#)
 READ, Befehl [378](#)
 READNEXT, Befehl [390](#)
 READPREV, Befehl [401](#)
 READQ TD, Befehl [408](#)
 READQ TS, Befehl [412](#)
 RECEIVE (z/OS Communications Server), Befehl [422](#)
 RECEIVE MAP MAPPINGDEV, Befehl [436](#)
 RECEIVE MAP, Befehl [433](#)
 RECEIVE PARTN, Befehl [438](#)
 RECEIVE, Befehl (ohne z/OS Communications Server) [429](#)
 RETRIEVE, Befehl [455](#)
 SPOOLREAD, Befehl [560](#)
 WEB CONVERSE, Befehl [652](#)
 WEB RECEIVE, Befehl (Client) [694](#)
 WEB RECEIVE, Befehl (Server) [687](#)
 INTOCCSID, Option
 GET CONTAINER (CHANNEL), Befehl [237](#)
 GET64 CONTAINER, Befehl [261](#)
 INTOCODEPAGE, Option
 GET CONTAINER (CHANNEL), Befehl [238](#)
 GET64 CONTAINER, Befehl [261](#)
 INVALIDCOUNT, Option
 VERIFY PASSWORD, Befehl [620](#)
 VERIFY PHRASE, Befehl [624](#)
 INVERRTERM, Bedingung
 ROUTE, Befehl [476](#)
 INVITE, Option
 GDS SEND, Befehl [230](#)
 SEND (ohne z/OS Communications Server), Befehl [498](#)
 SEND (z/OS Communications Server), Befehl [494](#)
 INVLDC, Bedingung
 ROUTE, Befehl [476](#)
 SEND CONTROL, Befehl [504](#)
 SEND MAP, Befehl [511](#)
 INVLDLDC, Bedingung (*Forts.*)
 SEND TEXT, Befehl [525](#)
 INVMPSTZ, Bedingung
 RECEIVE MAP MAPPINGDEV, Befehl [437](#)
 RECEIVE MAP, Befehl [434](#)
 SEND MAP MAPPINGDEV, Befehl [515](#)
 SEND MAP, Befehl [512](#)
 INVOKE APPLICATION, Befehl [278](#)
 INVOKE SERVICE, Befehl [282](#)
 INVOKE WEBSERVICE, Befehl [282](#), [287](#)
 INVOKINGPROG, Option
 ASSIGN, Befehl [50](#)
 INVPARTN, Bedingung
 RECEIVE MAP, Befehl [434](#)
 RECEIVE PARTN, Befehl [439](#)
 SEND CONTROL, Befehl [504](#)
 SEND MAP, Befehl [512](#)
 SEND TEXT NOEDIT, Befehl [531](#)
 SEND TEXT, Befehl [525](#)
 INVPARTNSET, Bedingung
 SEND PARTNSET, Befehl [519](#)
 INVREQ, Bedingung
 ACQUIRE, Befehl [30](#)
 ADD SUBEVENT, Befehl [32](#)
 ALLOCATE (APPC), Befehl [37](#)
 ALLOCATE (LUTYPE6.1), Befehl [39](#)
 ALLOCATE (MRO), Befehl [40](#)
 ASSIGN, Befehl [57](#)
 BIF DIGEST, Befehl [61](#)
 CANCEL, Befehl (BTS) [70](#)
 CHANGE PASSWORD, Befehl [74](#)
 CHANGE PHRASE, Befehl [72](#)
 CHANGE TASK, Befehl [75](#)
 CHECK ACQPROCESS, Befehl [77](#)
 CHECK ACTIVITY, Befehl [80](#)
 CHECK TIMER, Befehl [81](#)
 CONNECT PROCESS, Befehl [83](#)
 CONVERSE (z/OS Communications Server), Befehl [97](#)
 CONVERTTIME, Befehl [104](#)
 DEFINE ACTIVITY, Befehl [106](#)
 DEFINE COMPOSITE EVENT, Befehl [109](#)
 DEFINE INPUT EVENT, Befehl [113](#)
 DEFINE PROCESS, Befehl [115](#)
 DEFINE TIMER, Befehl [118](#)
 DELAY, Befehl [118](#)
 DELETE ACTIVITY, Befehl [130](#)
 DELETE CONTAINER (BTS), Befehl [132](#)
 DELETE CONTAINER (CHANNEL), Befehl [134](#)
 DELETE COUNTER, Befehl [109](#), [134](#), [364](#)
 DELETE DCOUNTER, Befehl [109](#)
 DELETE EVENT, Befehl [137](#)
 DELETE TIMER, Befehl [137](#)
 DELETE, Befehl [121](#)
 DELETEDQ TD, Befehl [138](#)
 DELETEDQ TS, Befehl [140](#)
 DEQ, Befehl [142](#)
 DOCUMENT RETRIEVE, Befehl [151](#)
 DUMP TRANSACTION, Befehl [158](#)
 ENDBR, Befehl [160](#)
 ENQ, Befehl [167](#)
 ENTER TRACENUM, Befehl [169](#)
 EXTRACT ATTACH (LUTYPE6.1), Befehl [172](#)
 EXTRACT ATTACH (MRO), Befehl [175](#)
 EXTRACT ATTRIBUTES (APPC), Befehl [176](#)

INVREQ, Bedingung (Forts.)

[EXTRACT CERTIFICATE, Befehl 181](#)
[EXTRACT PROCESS, Befehl 182](#)
[EXTRACT TCPIP, Befehl 187](#)
[EXTRACT TCT, Befehl 188](#)
[FORCE TIMER, Befehl 197](#)
[FORMATTIME, Befehl 202](#)
[FREE \(APPC\), Befehl 204](#)
[FREE \(LUTYPE6.1\), Befehl 205](#)
[FREE \(MRO\), Befehl 206](#)
[FREEMAIN, Befehl 208, 210](#)
[GET CONTAINER \(BTS\), Befehl 235](#)
[GET CONTAINER \(CHANNEL\), Befehl 239](#)
[GET64 CONTAINER, Befehl 263](#)
[GETMAIN64, Befehl 251](#)
[HANDLE AID, Befehl 267](#)
[INQUIRE EVENT, Befehl 275](#)
[INQUIRE TIMER, Befehl 278](#)
[INVOKE APPLICATION, Befehl 281](#)
[ISSUE ABEND, Befehl 288](#)
[ISSUE ABORT, Befehl 290](#)
[ISSUE ADD, Befehl 292](#)
[ISSUE CONFIRMATION, Befehl 293](#)
[ISSUE END, Befehl 298](#)
[ISSUE ENDFILE, Befehl 299](#)
[ISSUE ENDOUTPUT, Befehl 299](#)
[ISSUE EODS, Befehl 300](#)
[ISSUE ERASE, Befehl 301](#)
[ISSUE ERASEAUP, Befehl 303](#)
[ISSUE ERROR, Befehl 304](#)
[ISSUE NOTE, Befehl 306](#)
[ISSUE PASS, Befehl 308](#)
[ISSUE PREPARE, Befehl 309](#)
[ISSUE PRINT, Befehl 310](#)
[ISSUE QUERY, Befehl 311](#)
[ISSUE RECEIVE, Befehl 313](#)
[ISSUE REPLACE, Befehl 315](#)
[ISSUE SEND, Befehl 317](#)
[ISSUE SIGNAL \(APPC\), Befehl 319](#)
[ISSUE WAIT, Befehl 321](#)
[LINK ACQPROCESS, Befehl 332](#)
[LINK ACTIVITY, Befehl 335](#)
[LINK, Befehl 327](#)
[LOAD, Befehl 338](#)
[MONITOR, Befehl 341](#)
[MOVE CONTAINER \(BTS\), Befehl 344](#)
[MOVE CONTAINER \(CHANNEL\), Befehl 347](#)
[POP HANDLE, Befehl 348](#)
[POST, Befehl 351](#)
[PURGE MESSAGE, Befehl 352](#)
[PUT CONTAINER \(BTS\), Befehl 355](#)
[PUT CONTAINER \(CHANNEL\), Befehl 359](#)
[PUT64 CONTAINER, Befehl 363](#)
[QUERY SECURITY, Befehl 372](#)
[READ, Befehl 382](#)
[READNEXT, Befehl 394](#)
[READPREV, Befehl 405](#)
[READQ TD, Befehl 409](#)
[READQ TS, Befehl 413](#)
[RECEIVE MAP MAPPINGDEV, Befehl 437](#)
[RECEIVE MAP, Befehl 435](#)
[RECEIVE PARTN, Befehl 439](#)
[RECEIVE, Befehl \(ohne z/OS Communications Server\) 431](#)

INVREQ, Bedingung (Forts.)

[RELEASE, Befehl 440](#)
[REMOVE SUBEVENT, Befehl 442](#)
[REQUEST PASSTICKET, Befehl 444, 445](#)
[RESET ACQPROCESS, Befehl 447](#)
[RESET ACTIVITY, Befehl 448](#)
[RESETBR, Befehl 452](#)
[RESUME, Befehl 454](#)
[RETRIEVE REATTACH EVENT, Befehl 459](#)
[RETRIEVE SUBEVENT, Befehl 460](#)
[RETRIEVE, Befehl 457](#)
[RETURN, Befehl 464](#)
[REWRITE, Befehl 469](#)
[ROUTE, Befehl 476](#)
[RUN, Befehl 480](#)
[SEND \(ohne z/OS Communications Server\), Befehl 500](#)
[SEND CONTROL, Befehl 505](#)
[SEND MAP MAPPINGDEV, Befehl 515](#)
[SEND MAP, Befehl 512](#)
[SEND PAGE, Befehl 518](#)
[SEND PARTNSET, Befehl 519](#)
[SEND TEXT MAPPED, Befehl 527](#)
[SEND TEXT NOEDIT, Befehl 531](#)
[SEND TEXT, Befehl 525](#)
[SIGNOFF, Befehl 533](#)
[SIGNON, Befehl 537, 541](#)
[SOAPFAULT ADD, Befehl 545](#)
[SOAPFAULT CREATE, Befehl 549](#)
[SOAPFAULT DELETE, Befehl 550](#)
[SPOOLCLOSE, Befehl 551](#)
[SPOOLOPEN INPUT, Befehl 553](#)
[SPOOLOPEN OUTPUT, Befehl 558](#)
[SPOOLREAD, Befehl 561](#)
[SPOOLWRITE, Befehl 563](#)
[START ATTACH, Befehl 575](#)
[START BREXIT, Befehl 576](#)
[START TRANSID \(CHANNEL\), Befehl 579](#)
[START, Befehl 565](#)
[STARTBR, Befehl 587](#)
[STARTBROWSE EVENT, Befehl 595](#)
[SUSPEND \(BTS\), Befehl 598](#)
[SYNCPOINT ROLLBACK, Befehl 601](#)
[SYNCPOINT, Befehl 599](#)
[TEST EVENT, Befehl 601](#)
[UNLOCK, Befehl 613](#)
[VERIFY PASSWORD, Befehl 620, 624](#)
[VERIFY TOKEN, Befehl 628](#)
[WAIT CONVID, Befehl 630](#)
[WAIT EVENT, Befehl 631](#)
[WAIT EXTERNAL, Befehl 633](#)
[WAIT TERMINAL, Befehl 639](#)
[WAITCICS, Befehl 641](#)
[WEB CONVERSE, Befehl 656](#)
[WEB ENDBROWSE FORMFIELD, Befehl 659](#)
[WEB ENDBROWSE HTTPHEADER, Befehl 659](#)
[WEB ENDBROWSE QUERYPARM, Befehl 660](#)
[WEB EXTRACT oder EXTRACT WEB, Befehl 192, 664](#)
[WEB OPEN, Befehl 670](#)
[WEB PARSE URL, Befehl 674](#)
[WEB READ FORMFIELD, Befehl 675](#)
[WEB READ HTTPHEADER, Befehl 677](#)
[WEB READ QUERYPARM, Befehl 679](#)
[WEB READNEXT FORMFIELD, Befehl 681](#)
[WEB READNEXT HTTPHEADER, Befehl 682](#)

INVREQ, Bedingung (Forts.)

WEB READNEXT QUERYPARM, Befehl [684](#)
WEB RECEIVE, Befehl (Client) [697](#)
WEB RECEIVE, Befehl (Server) [690](#)
WEB RETRIEVE, Befehl [699](#)
WEB SEND, Befehl (Client) [717](#)
WEB SEND, Befehl (Server) [706](#)
WEB STARTBROWSE FORMFIELD, Befehl [721](#)
WEB STARTBROWSE HTTPHEADER, Befehl [722](#)
WEB STARTBROWSE QUERYPARM, Befehl [724](#)
WEB WRITE HTTPHEADER, Befehl [726](#)
WRITE JOURNALNAME, Befehl [736](#)
WRITE OPERATOR, Befehl [740](#)
WRITE, Befehl [731](#)
WRITEQ TD, Befehl [742](#)
WRITEQ TS, Befehl [746](#)
WSACONTEXT BUILD, Befehl [751](#)
XCTL, Befehl [761](#)

IOERR, Bedingung

ACQUIRE, Befehl [30](#)
CANCEL, Befehl (BTS) [70](#)
CHECK ACTIVITY, Befehl [80](#)
CHECK TIMER, Befehl [81](#)
DEFINE ACTIVITY, Befehl [106](#)
DEFINE PROCESS, Befehl [115](#)
DELETE ACTIVITY, Befehl [130](#)
DELETE CONTAINER (BTS), Befehl [132](#)
DELETE, Befehl [121](#)
DUMP TRANSACTION, Befehl [158](#)
ENDBR, Befehl [160](#)
GET CONTAINER (BTS), Befehl [235](#)
GETNEXT ACTIVITY, Befehl [253](#)
GETNEXT PROCESS, Befehl [257](#)
INQUIRE CONTAINER, Befehl [273](#)
INQUIRE EVENT, Befehl [275](#)
INQUIRE TIMER, Befehl [278](#)
LINK ACQPROCESS, Befehl [332](#)
LINK ACTIVITY, Befehl [336](#)
MOVE CONTAINER (BTS), Befehl [344](#)
PUT CONTAINER (BTS), Befehl [355](#)
READ, Befehl [383](#)
READNEXT, Befehl [395](#)
READPREV, Befehl [406](#)
READQ TD, Befehl [409](#)
READQ TS, Befehl [413](#)
RESET ACQPROCESS, Befehl [447](#)
RESET ACTIVITY, Befehl [448](#)
RESETBR, Befehl [452](#)
RESUME, Befehl [454](#)
RETRIEVE, Befehl [457](#)
REWRITE, Befehl [470](#)
RUN, Befehl [480](#)
START, Befehl [565](#)
STARTBR, Befehl [588](#)
STARTBROWSE CONTAINER, Befehl [593](#)
STARTBROWSE EVENT, Befehl [595](#)
STARTBROWSE PROCESS, Befehl [596](#)
SUSPEND (BTS), Befehl [599](#)
UNLOCK, Befehl [614](#)
WAIT JOURNALNAME, Befehl [636](#)
WEB CONVERSE, Befehl [658](#)
WEB EXTRACT oder EXTRACT WEB (Client), Befehl [192](#),
[665](#)
WEB OPEN, Befehl [670](#)

IOERR, Bedingung (Forts.)

WEB RECEIVE, Befehl (Client) [698](#)
WEB SEND, Befehl (Client) [719](#)
WEB SEND, Befehl (Server) [708](#)
WRITE JOURNALNAME, Befehl [737](#)
WRITE, Befehl [731](#)
WRITEQ TD, Befehl [742](#)
WRITEQ TS, Befehl [746](#)
ISCINVREQ, Bedingung
CANCEL, Befehl [69](#)
DELETE, Befehl [121](#)
DELETEQ TD, Befehl [138](#)
DELETEQ TS, Befehl [140](#)
ENDBR, Befehl [161](#)
READ, Befehl [383](#)
READNEXT, Befehl [396](#)
READPREV, Befehl [406](#)
READQ TD, Befehl [410](#)
READQ TS, Befehl [413](#)
RESETBR, Befehl [452](#)
REWRITE, Befehl [470](#)
START TRANSID (CHANNEL), Befehl [579](#)
START, Befehl [565](#)
STARTBR, Befehl [588](#)
UNLOCK, Befehl [614](#)
WRITE, Befehl [732](#)
WRITEQ TD, Befehl [742](#)
WRITEQ TS, Befehl [746](#)
ISSUE ABEND, Befehl [287](#)
ISSUE ABORT, Befehl [288](#)
ISSUE ADD, Befehl [290](#)
ISSUE CONFIRMATION, Befehl [292](#)
ISSUE COPY (3270, logisch), Befehl [293](#)
ISSUE COPY, Befehl
 allgemeine Informationen [836](#)
ISSUE DISCONNECT (LUTYPE6.1), Befehl [296](#)
ISSUE DISCONNECT (Standard), Befehl [294](#)
ISSUE DISCONNECT, Befehl
 allgemeine Informationen [833](#)
ISSUE END, Befehl [296](#)
ISSUE ENDFILE, Befehl [298](#)
ISSUE ENDOUTPUT, Befehl [299](#)
ISSUE EODS, Befehl [299](#)
ISSUE ERASE, Befehl [300](#)
ISSUE ERASEAUP, Befehl
 allgemeine Informationen [837](#)
ISSUE ERROR, Befehl [303](#)
ISSUE LOAD, Befehl [304](#)
ISSUE NOTE, Befehl [305](#)
ISSUE PASS, Befehl [307](#)
ISSUE PREPARE, Befehl [308](#)
ISSUE PRINT, Befehl
 allgemeine Informationen [836](#)
ISSUE QUERY, Befehl [311](#)
ISSUE RECEIVE, Befehl [312](#)
ISSUE REPLACE, Befehl [313](#)
ISSUE RESET, Befehl [316](#)
ISSUE SEND, Befehl [316](#)
ISSUE SIGNAL (APPC), Befehl [318](#)
ISSUE SIGNAL (LUTYPE6.1), Befehl [319](#)
ISSUE SIGNAL, Befehl
 allgemeine Informationen [833](#)
ISSUE WAIT, Befehl [320](#)
ISSUER, Option

ISSUER, Option (*Forts.*)
 EXTRACT CERTIFICATE, Befehl [180](#)
 ISUSERID, Option
 VERIFY TOKEN, Befehl [627](#)
 ITEM, Option
 READQ TS, Befehl [412](#)
 WRITEQ TS, Befehl [744](#)
 ITEMERR, Bedingung
 READQ TS, Befehl [413](#)
 WRITEQ TS, Befehl [746](#)
 IUTYPE, Option
 BUILD ATTACH (LUTYPE6.1), Befehl [63](#)
 BUILD ATTACH (MRO), Befehl [66](#)
 EXTRACT ATTACH (LUTYPE6.1), Befehl [171](#)
 EXTRACT ATTACH (MRO), Befehl [174](#)

J

JIDERR, Bedingung
 WRITE JOURNALNAME, Befehl [737](#)
 JIDERR, Option
 WAIT JOURNALNAME, Befehl [636](#)
 JOURNAL, Befehl [322](#)
 Journalführung, Befehle [22](#)
 JOURNALNAME, Option
 WAIT JOURNALNAME, Befehl [636](#)
 WRITE JOURNALNAME, Befehl [735](#)
 Journalsatzerstellung [734](#)
 Journalsteuerung
 Erstellen eines Journalsatzes [322](#)
 JTYPEID, Option
 WRITE JOURNALNAME, Befehl [736](#)
 JUSFIRST, Option
 SEND TEXT, Befehl [522](#)
 JUSLAST, Option
 SEND TEXT, Befehl [522](#)
 JUSTIFY, Operand
 DFHMDI [851](#)
 DFHMDI [862](#)
 JUSTIFY, Option
 SEND TEXT, Befehl [522](#)

K

Kanalbefehle
 DELETE CHANNEL [130](#)
 DELETE CONTAINER (CHANNEL) [133](#)
 ENDBROWSE CONTAINER (CHANNEL) [162](#)
 GET CONTAINER (CHANNEL) [236](#)
 GET64 CONTAINER [259](#)
 GETNEXT CONTAINER (CHANNEL) [254](#)
 MOVE CONTAINER (CHANNEL) [345](#)
 Option CHANNEL des Befehls RETURN [460](#)
 Option CHANNEL des Befehls XCTL [759](#)
 PUT CONTAINER (CHANNEL) [355](#)
 PUT64 CONTAINER [359](#)
 QUERY CHANNEL [363](#)
 START CHANNEL [579](#)
 STARTBROWSE CONTAINER (CHANNEL) [593](#)
 Kanäle
 ASSIGN, Befehl [47](#)
 Kanäle als große COMMAREAs [130](#), [133](#), [236](#), [345](#), [355](#),
[363](#), [460](#), [579](#), [759](#)

Katakana- und englische Zeichen gemischt [99](#), [438](#)
 Katakana-Terminals
 Befehl RECEIVE (ohne z/OS Communications Server) [429](#)
 CONVERSE (3270 logisch), Befehl [93](#)
 CONVERSE (LUTYPE2/LUTYPE3), Befehl [93](#)
 CONVERSE, Befehl (3270-Anzeige) [99](#)
 CONVERSE, Befehl (3600-BTAM) [99](#)
 CONVERSE, Befehl (3735) [99](#)
 CONVERSE, Befehl (3740) [99](#)
 CONVERSE, Befehl (System/3) [99](#)
 CONVERSE, Befehl (System/7) [99](#)
 RECEIVE (z/OS Communications Server), Befehl [421](#)
 RECEIVE MAP, Befehl [433](#)
 RECEIVE PARTN, Befehl [438](#)
 SEND (ohne z/OS Communications Server), Befehl [497](#)
 KATAKANA, Option
 ASSIGN, Befehl [51](#)
 KEEP, Option
 SPOOLCLOSE, Befehl [551](#)
 Kennwortphrase
 WEB CONVERSE, Befehl [650](#)
 KEYLENGTH, Option
 DELETE, Befehl [121](#)
 ISSUE ERASE, Befehl [301](#)
 ISSUE REPLACE, Befehl [314](#)
 READ, Befehl [378](#)
 READNEXT, Befehl [390](#)
 READPREV, Befehl [401](#)
 RESETBR, Befehl [450](#)
 STARTBR, Befehl [585](#)
 WRITE, Befehl [728](#)
 KEYNUMBER, Option
 ISSUE ERASE, Befehl [301](#)
 ISSUE REPLACE, Befehl [314](#)
 Klassendeskriptortabelle (CDT) [370](#)
 Konsolenunterstützung, Befehle [20](#)
 Konstanten
 AID-Werte, DFHAID [846](#)
 Attributwerte, DFHBMSA [841](#)
 Druckersteuerwerte, DFHBMSA [841](#)
 für 3270-Attribute [841](#)
 für Druckerformatsteuerzeichen [841](#)
 für MSR-Steuerwerte [845](#)
 MSR-Steuerung, DFHtex read [845](#)
 zum Untersuchen eines EIBAID-Felds [846](#)
 Kontextwechsel
 Beschreibung [331](#), [334](#), [477](#)
 Konvertierter Code [11](#)
 Kopieren von angezeigten Informationen [836](#)

L

L40, L64 oder L80, Optionen
 SEND CONTROL, Befehl [503](#)
 SEND MAP, Befehl [509](#)
 SEND TEXT NOEDIT, Befehl [529](#)
 SEND TEXT, Befehl [523](#)
 LABEL, Option
 HANDLE ABEND, Befehl [264](#)
 Laden von Programmen, Tabellen oder Masken [337](#)
 Landessprachencodes [831](#)
 LANG, Operand
 DFHMSD [871](#)

LANGINUSE, Option
 ASSIGN [51](#)
 SIGNON TOKEN, Befehl [540](#)
 SIGNON, Befehl [535](#)
 LANGUAGECODE, Option
 SIGNON TOKEN, Befehl [540](#)
 SIGNON, Befehl [535](#)
 LAST, Option
 GDS SEND, Befehl [231](#)
 SEND (ohne z/OS Communications Server), Befehl [498](#)
 SEND (z/OS Communications Server), Befehl [494](#)
 SEND CONTROL, Befehl [503](#)
 SEND MAP, Befehl [508](#)
 SEND PAGE, Befehl [517](#)
 SEND TEXT MAPPED, Befehl [526](#)
 SEND TEXT NOEDIT, Befehl [529](#)
 SEND TEXT, Befehl [522](#)
 LAST, Wert
 DFHMDI [862](#)
 LASTUSETIME, Option
 VERIFY PASSWORD, Befehl [620](#)
 VERIFY PHRASE, Befehl [624](#)
 LDC, Operand
 DFHMSD [871](#)
 LDC, Option
 CONVERSE (z/OS Communications Server), Befehl [94](#)
 ROUTE, Befehl [474](#)
 SEND (z/OS Communications Server), Befehl [494](#)
 SEND CONTROL, Befehl [503](#)
 SEND MAP, Befehl [509](#)
 SEND TEXT, Befehl [523](#)
 LDCMNEM, Option
 ASSIGN, Befehl [51](#)
 LDCNUM, Option
 ASSIGN, Befehl [51](#)
 LEAVEKB, Option
 CONVERSE (Nicht-z/OS Communications Server), Befehl [100](#)
 RECEIVE, Befehl (ohne z/OS Communications Server) [429](#)
 SEND (ohne z/OS Communications Server), Befehl [499](#)
 LEFT, Wert
 DFHMDI [851](#)
 DFHMDI [862](#)
 Leistung, Anwendung, überwachen [340](#)
 LENGERR, Bedingung
 BIF DEEDIT, Befehl [60](#)
 BIF DIGEST, Befehl [61](#)
 CHANGE PHRASE, Befehl [73](#)
 CONNECT PROCESS, Befehl [83](#)
 CONVERSE (Nicht-z/OS Communications Server), Befehl [102](#)
 CONVERSE (z/OS Communications Server), Befehl [97](#)
 DEQ, Befehl [142](#)
 DOCUMENT RETRIEVE, Befehl [151](#)
 ENQ, Befehl [167](#)
 ENTER TRACENUM, Befehl [169](#)
 EXTRACT PROCESS, Befehl [182](#)
 EXTRACT TCPIP, Befehl [187](#)
 GET CONTAINER (BTS), Befehl [235](#)
 GET CONTAINER (CHANNEL), Befehl [240](#)
 GET64 CONTAINER, Befehl [263](#)
 GETMAIN, Befehl [248](#)
 GETMAIN64, Befehl [251](#)
 LENGERR, Bedingung (Forts.)
 INVOKE APPLICATION, Befehl [281](#)
 ISSUE COPY (3270, logisch), Befehl [294](#)
 ISSUE PASS, Befehl [308](#)
 ISSUE RECEIVE, Befehl [313](#)
 LINK, Befehl [328](#)
 LOAD, Befehl [339](#)
 PUT CONTAINER (CHANNEL), Befehl [359](#)
 PUT64 CONTAINER, Befehl [363](#)
 QUERY SECURITY, Befehl [372](#)
 READ, Befehl [384](#)
 READNEXT, Befehl [396](#)
 READPREV, Befehl [406](#)
 READQ TD, Befehl [410](#)
 READQ TS, Befehl [413](#)
 RECEIVE (z/OS Communications Server), Befehl [424](#)
 RECEIVE PARTN, Befehl [439](#)
 RECEIVE, Befehl (ohne z/OS Communications Server) [431](#)
 RETRIEVE, Befehl [457](#)
 RETURN, Befehl [464](#)
 REWRITE, Befehl [470](#)
 SEND (ohne z/OS Communications Server), Befehl [500](#)
 SEND (z/OS Communications Server), Befehl [495](#)
 SEND TEXT, Befehl [525](#)
 SIGNAL EVENT, Befehl [532](#)
 SIGNON, Befehl [537](#), [542](#)
 SOAPFAULT CREATE, Befehl [545](#), [549](#)
 SPOOL OPEN OUTPUT, Befehl [558](#)
 SPOOL READ, Befehl [561](#)
 SPOOL WRITE, Befehl [563](#)
 START ATTACH, Befehl [575](#)
 START BREXIT, Befehl [576](#)
 START, Befehl [565](#)
 VERIFY PHRASE, Befehl [624](#)
 VERIFY TOKEN, Befehl [629](#)
 WEB CONVERSE, Befehl [657](#)
 WEB EXTRACT oder EXTRACT WEB, Befehl [192](#), [665](#)
 WEB OPEN, Befehl [671](#)
 WEB PARSE URL, Befehl [674](#)
 WEB READ FORMFIELD, Befehl [676](#)
 WEB READ HTTPHEADER, Befehl [677](#)
 WEB READ QUERYPARM, Befehl [680](#)
 WEB READNEXT FORMFIELD, Befehl [681](#)
 WEB READNEXT HTTPHEADER, Befehl [683](#)
 WEB READNEXT QUERYPARM, Befehl [684](#)
 WEB RECEIVE, Befehl (Client) [698](#)
 WEB RECEIVE, Befehl (Server) [691](#)
 WEB SEND, Befehl (Client) [708](#), [719](#)
 WEB STARTBROWSE FORMFIELD, Befehl [721](#)
 WEB STARTBROWSE QUERYPARM, Befehl [724](#)
 WEB WRITE HTTPHEADER, Befehl [727](#)
 WRITE JOURNALNAME, Befehl [737](#)
 WRITE OPERATOR, Befehl [741](#)
 WRITE, Befehl [732](#)
 WRITEQ TD, Befehl [742](#)
 WRITEQ TS, Befehl [746](#)
 XCTL, Befehl [762](#)
 LENGTH, Operand
 DFHMDI [851](#)
 LENGTH, Option
 BIF DEEDIT, Befehl [60](#)
 DEQ, Befehl [141](#)
 DOCUMENT RETRIEVE, Befehl [151](#)

LENGTH, Option (*Forts.*)

- DOCUMENT SET, Befehl [153](#)
- DUMP TRANSACTION, Befehl [156](#)
- ENQ, Befehl [166](#)
- EXTRACT CERTIFICATE, Befehl [180](#)
- EXTRACT LOGONMSG, Befehl [181](#)
- GETMAIN, Befehl [247](#)
- integrierte Funktion [60](#)
- INVOKE APPLICATION, Befehl [279](#)
- ISSUE ADD, Befehl [291](#)
- ISSUE PASS, Befehl [307](#)
- ISSUE RECEIVE, Befehl [312](#)
- ISSUE REPLACE, Befehl [314](#)
- ISSUE SEND, Befehl [317](#)
- LINK, Befehl [325](#)
- LOAD, Befehl [338](#)
- READ, Befehl [378](#)
- READNEXT, Befehl [391](#)
- READPREV, Befehl [401](#)
- READQ TD, Befehl [408](#)
- READQ TS, Befehl [412](#)
- RECEIVE (z/OS Communications Server), Befehl [422](#)
- RECEIVE MAP MAPPINGDEV, Befehl [436](#)
- RECEIVE MAP, Befehl [433](#)
- RECEIVE PARTN, Befehl [438](#)
- RECEIVE, Befehl (ohne z/OS Communications Server) [429](#)
- RETRIEVE, Befehl [456](#)
- RETURN, Befehl [463](#)
- REWRITE, Befehl [468](#)
- SEND (ohne z/OS Communications Server), Befehl [499](#)
- SEND (z/OS Communications Server), Befehl [494](#)
- SEND MAP MAPPINGDEV, Befehl [515](#)
- SEND MAP, Befehl [509](#)
- SEND TEXT MAPPED, Befehl [527](#)
- SEND TEXT NOEDIT, Befehl [529](#)
- SEND TEXT, Befehl [523](#)
- Standard (Assemblersprache) [8](#)
- Standard (C) [5](#)
- Standard (PL/I) [7](#)
- START ATTACH, Befehl [575](#)
- START, Befehl [571](#)
- Vollwortalternative für Länge (FLENGTH) [833](#)
- WEB RECEIVE (Server), Befehl [688](#)
- WEB RECEIVE, Befehl (Client) [694](#)
- WEB SEND, Befehl (Server) [704](#)
- WRITE, Befehl [728](#)
- WRITEQ TD, Befehl [741](#)
- WRITEQ TS, Befehl [744](#)
- XCTL, Befehl [761](#)

LENGTH, Wert

- DFHMDI [862](#)
- DFHMSD [871](#)

LENGTHLIST, Option

- DUMP TRANSACTION, Befehl [156](#)

Lesen von Datensätzen

- aus Terminal oder logischer Einheit [833](#)
- aus Warteschlange für temporären Speicher [411](#)
- Batch Data Interchange [312](#)
- Dateisteuerung [374](#)
- nächsten suchen [387](#)
- vorherigen suchen (VSAM) [398](#)

LEVEL, Option

- GETNEXT ACTIVITY, Befehl [252](#)

LIGHTPEN, Option

- HANDLE AID, Befehl [265](#)

LINE, Operand

- DFHMDI [862](#)

LINE, Option

- SPOOLWRITE, Befehl [563](#)

LINEADDR, Option

- CONVERSE (Nicht-z/OS Communications Server), Befehl [100](#)

- SEND (ohne z/OS Communications Server), Befehl [499](#)

LINK ACQPROCESS, Befehl [331](#)

LINK ACTIVITY, Befehl [334](#)

LINKLEVEL, Option

- ASSIGN, Befehl [51](#)

LIST, Option

- ROUTE, Befehl [475](#)

Literalkonstanten [7](#)

LLID, Option

- GDS RECEIVE, Befehl [228](#)

LOAD, Befehl [337](#)

LOADING, Bedingung

- DELETE, Befehl [121](#)

- READ, Befehl [384](#)

- READNEXT, Befehl [396](#)

- STARTBR, Befehl [588](#)

- WRITE, Befehl [732](#)

LOCAL [201](#)

LOCALCCSID, Option

- ASSIGN, Befehl [51](#)

LOCALITY, Option

- EXTRACT CERTIFICATE, Befehl [180](#)

LOCALITYLEN, Option

- EXTRACT CERTIFICATE, Befehl [180](#)

LOCATION(LOC24), Option

- GETMAIN64, Befehl [251](#)

LOCKED, Bedingung

- ACQUIRE, Befehl [30](#)

- CANCEL, Befehl (BTS) [70](#)

- CHECK ACTIVITY, Befehl [80](#)

- DELETE ACTIVITY, Befehl [130](#)

- DELETE CONTAINER (BTS), Befehl [133](#)

- DELETE, Befehl [121](#)

- DELETEQ TD, Befehl [138](#)

- DELETEQ TS, Befehl [140](#)

- GET CONTAINER (BTS), Befehl [235](#)

- LINK ACTIVITY, Befehl [336](#)

- MOVE CONTAINER (BTS), Befehl [345](#)

- PUT CONTAINER (BTS), Befehl [355](#)

- READ, Befehl [384](#)

- READNEXT, Befehl [396](#)

- READPREV, Befehl [406](#)

- READQ TD, Befehl [410](#)

- RESET ACQPROCESS, Befehl [447](#)

- RESET ACTIVITY, Befehl [448](#)

- RESUME, Befehl [454](#)

- REWRITE, Befehl [471](#)

- RUN, Befehl [480](#)

- SUSPEND (BTS), Befehl [599](#)

- WRITE, Befehl [732](#)

- WRITEQ TD, Befehl [742](#)

- WRITEQ TS, Befehl [746](#)

Logische 3650- bzw. 3680-Einheit mit vollem Funktionsumfang

Befehl SEND [491](#)

Logische 3650- bzw. 3680-Einheit mit vollem Funktionsumfang (*CONVERSE*)
 RECEIVE, Befehl [421](#)
 Logische 3650-Einheit
 RECEIVE, Befehl [419](#)
 Logische 3770-Einheit mit vollem Funktionsumfang
 RECEIVE, Befehl [421](#)
 SEND, Befehl [491](#)
 Logische 3790-Einheit mit vollem Funktionsumfang [421](#), [491](#)
 Logische Nachrichten, BMS
 logische Nachricht abschließen [516](#)
 logische Nachricht bereinigen [352](#)
 logische Nachricht weiterleiten [472](#)
 vollständige BMS
 Befehl ROUTE [472](#)
 Logischer Einheitencode (LDC) [88](#), [487](#)
 LOGMESSAGE, Option
 QUERY SECURITY, Befehl [368](#)
 LOGMODE, Option
 ISSUE PASS, Befehl [307](#)
 LOGONLOGMODE, Option
 ISSUE PASS [308](#)
 Löschen eines geladenen Programms [439](#)
 Löschen von Aktivitäten [129](#)
 Löschen von Daten
 benannter Zähler [134](#)
 temporäre Speicherwarteschlangen [139](#)
 Warteschlangen für transiente Daten [137](#)
 Löschen von Datensätzen
 Batch Data Interchange-Datensätze [300](#)
 LU (logische Einheit)
 3270-Anzeige, LUTYPE2 [84](#), [415](#), [483](#)
 3270-Anzeige, LUTYPE3 [415](#), [483](#)
 3270, Informationsanzeigesystem [87](#), [293](#), [417](#), [485](#)
 3270, SCS-Drucker [86](#), [485](#)
 3600 (3601) [87](#), [418](#), [486](#)
 3600 (3614) [88](#), [418](#), [487](#)
 3600-Pipeline [417](#), [486](#)
 3650, Hostdialog (3270) [89](#)
 3650, Hostdialog (3653) [90](#)
 3650, Interpreter [89](#), [299](#), [304](#), [419](#), [488](#)
 3770, Stapelverarbeitung [91](#), [420](#), [490](#)
 3790 mit vollem Funktionsumfang [421](#), [491](#)
 3790- bzw. 3270-Anzeige [92](#), [428](#), [491](#)
 3790- bzw. 3270-Drucker [492](#)
 3790-SCS-Drucker [491](#)
 3790, mit vollem Funktionsumfang oder Anfrage [92](#)
 Daten lesen aus [312](#), [833](#)
 Daten schreiben [290](#)
 Daten schreiben in [833](#)
 Dialog mit (CONVERSE) [833](#)
 interaktiv [91](#)
 mit 3650- bzw. 3270-Hostdialog [488](#)
 mit 3650- bzw. 3653-Hostdialog [489](#)
 mit 3650- bzw. 3680-Hostbefehlsprozessor [489](#)
 Stapelverarbeitung [91](#), [420](#), [490](#)
 LUNAME, Option
 ISSUE PASS, Befehl [308](#)
 LUTYPE2, logische Einheit mit 3270-Anzeige [84](#), [415](#), [483](#)
 LUTYPE3, logische Einheit mit 3270-Anzeige [415](#), [483](#)
 LUTYPE4
 logische Einheit [85](#), [416](#), [484](#)
 LUTYPE6.1, logische Einheit
 Daten empfangen [416](#)
 LUTYPE6.1, logische Einheit (*Forts.*)
 Daten senden [484](#)
 in Sitzung mit LU-Typ 6.1 (LUTYPE6.1) kommunizieren [86](#)
 Informationen abrufen [347](#)
 Namen aus 8 Zeichen in Namen aus 4 Zeichen konvertieren [187](#)
 Richtungswechsel anfordern [319](#)
 Sitzung anfordern [37](#)
 Verbindung trennen [296](#)
 Werte aus einem LUTYPE6.1-Header abrufen [169](#)
 Werte für LUTYPE6.1-Anschlussheader angeben [62](#)
 Werte für MRO-Anschlussheader angeben [65](#)

M

Magnetcodeleser (MSR) [845](#)
 MAIN, Option
 WRITEQ TS, Befehl [744](#)
 MAJORVERSION, Option
 INVOKE APPLICATION, Befehl [280](#)
 MAP, Option
 RECEIVE MAP MAPPINGDEV, Befehl [436](#)
 RECEIVE MAP, Befehl [433](#)
 SEND MAP MAPPINGDEV, Befehl [515](#)
 SEND MAP, Befehl [509](#)
 MAP, Wert
 DFHMSD [871](#)
 MAPATTS, Operand
 DFHMDI [862](#)
 DFHMSD [871](#)
 MAPCOLUMN, Option
 ASSIGN, Befehl [51](#)
 MAPFAIL, Bedingung
 RECEIVE MAP MAPPINGDEV, Befehl [437](#)
 RECEIVE MAP, Befehl [435](#)
 MAPHEIGHT, Option
 ASSIGN, Befehl [51](#)
 MAPLINE, Option
 ASSIGN, Befehl [51](#)
 MAPONLY, Option
 SEND MAP MAPPINGDEV, Befehl [515](#)
 SEND MAP, Befehl [509](#)
 MAPONLY, Wert
 DFHMDI [862](#)
 DFHMSD [871](#)
 MAPPINGDEV, Option
 RECEIVE MAP MAPPINGDEV, Befehl [436](#)
 SEND MAP MAPPINGDEV, Befehl [515](#)
 MAPSET, Option
 RECEIVE MAP MAPPINGDEV, Befehl [436](#)
 RECEIVE MAP, Befehl [434](#)
 SEND MAP MAPPINGDEV, Befehl [515](#)
 SEND MAP, Befehl [509](#)
 MAPSFX, Operand
 DFHPDI [881](#)
 MAPWIDTH, Option
 ASSIGN, Befehl [51](#)
 Masken laden [337](#)
 Maskendefinitionsmakro, BMS [849](#), [862](#)
 Maskengruppendefinitionsmakro (DFHMSD) [849](#), [871](#)
 MASSINSERT, Option
 WRITE, Befehl [729](#)
 MAXDATALEN, Option

MAXDATALEN, Option (*Forts.*)
 EXTRACT TCPIP, Befehl [185](#)
 MAXLENGTH, Option
 CONVERSE (Nicht-z/OS Communications Server), Befehl [100](#)
 CONVERSE (z/OS Communications Server), Befehl [95](#)
 GDS RECEIVE, Befehl [228](#)
 RECEIVE (z/OS Communications Server), Befehl [422](#)
 RECEIVE, Befehl (ohne z/OS Communications Server) [430](#)
 SPOOLREAD, Befehl [560](#)
 Vollwortalternative für MAXLENGTH [833](#)
 MAXIMUM, Option
 DEFINE COUNTER, Befehl [109](#)
 DEFINE DOUNTER, Befehl [109](#)
 QUERY COUNTER, Befehl [364](#)
 QUERY DOUNTER, Befehl [364](#)
 MAXLENGTH, Option
 CONVERSE (Nicht-z/OS Communications Server), Befehl [100](#)
 CONVERSE (z/OS Communications Server), Befehl [95](#)
 DOCUMENT RETRIEVE, Befehl [151](#)
 RECEIVE (z/OS Communications Server), Befehl [422](#)
 RECEIVE, Befehl (ohne z/OS Communications Server) [430](#)
 Vollwortalternative für Länge (MAXLENGTH) [833](#)
 WEB CONVERSE, Befehl [652](#)
 WEB RECEIVE, Befehl (Client) [695](#)
 WEB RECEIVE, Befehl (Server) [688](#)
 WRITE OPERATOR, Befehl [739](#)
 MAXLIFETIME, Option
 DEQ, Befehl [142](#)
 ENQ, Befehl [166](#)
 MAXPROCLN, Option
 EXTRACT PROCESS, Befehl [182](#)
 GDS EXTRACT PROCESS, Befehl [219](#)
 MCC, Option
 SPOOLOPEN OUTPUT, Befehl [556](#)
 MEDIATYPE, Option
 WEB CONVERSE, Befehl [649](#), [652](#)
 WEB RECEIVE, Befehl (Client) [695](#)
 WEB RECEIVE, Befehl (Server) [688](#)
 WEB SEND, Befehl (Client) [714](#)
 WEB SEND, Befehl (Server) [704](#)
 Mehrregionenbetrieb (MRO - Multi Region Operation), Befehle
 ALLOCATE [39](#)
 BUILD ATTACH [65](#)
 CONVERSE, Befehl [98](#)
 EXTRACT ATTACH [173](#)
 EXTRACT ATTRIBUTES [177](#)
 FREE [205](#)
 RECEIVE [425](#)
 SEND [496](#)
 METHOD, Option
 WEB CONVERSE, Befehl [650](#)
 WEB SEND, Befehl (Client) [715](#)
 METHODLENGTH, Option
 WEB EXTRACT oder EXTRACT WEB, Befehl [190](#), [663](#)
 MICROVERSION, Option [52](#)
 MILLISECONDS, Option
 FORMATIME, Befehl [200](#)
 MINIMUM, Option
 DEFINE COUNTER, Befehl [109](#)
 MINIMUM, Option (*Forts.*)
 DEFINE DOUNTER, Befehl [109](#)
 INVOKE APPLICATION, Befehl [280](#)
 QUERY COUNTER, Befehl [364](#)
 QUERY DOUNTER, Befehl [364](#)
 MINORVERSION, Option
 INVOKE APPLICATION, Befehl [280](#)
 MINUTES, Option
 DEFINE TIMER, Befehl [117](#)
 DELAY, Befehl [118](#)
 POST, Befehl [351](#)
 ROUTE, Befehl [475](#)
 START, Befehl [571](#)
 MMDDYY, Option
 FORMATIME, Befehl [200](#)
 MMDDYYYY, Option
 FORMATIME, Befehl [200](#)
 MODE, Operand
 DFHMSD [871](#)
 MODE, Option
 CHECK ACQPROCESS, Befehl [77](#)
 CHECK ACTIVITY, Befehl [79](#)
 INQUIRE ACTIVITYID, Befehl [271](#)
 Modellcodes für Terminals [58](#)
 MODENAME, Option
 GDS ALLOCATE, Befehl [212](#)
 Modi einer Aktivität
 ACTIVE [271](#)
 CANCELLING [271](#)
 COMPLETE [271](#)
 DORMANT [271](#)
 INITIAL [271](#)
 MONITOR, Befehl [340](#)
 MONTH, Option
 DEFINE TIMER, Befehl [117](#)
 MONTHOFYEAR, Option
 FORMATIME, Befehl [200](#)
 MOVE CONTAINER (BTS), Befehl [342](#)
 MOVE CONTAINER (CHANNEL), Befehl [345](#)
 MSR (Magnetcodeleser)
 DFHMSRCA [845](#)
 Steuerbytwerte und -konstanten [845](#)
 MSR, Option
 SEND CONTROL, Befehl [503](#)
 SEND MAP, Befehl [510](#)
 SEND TEXT, Befehl [523](#)
 MSRCONTROL, Option
 ASSIGN, Befehl [52](#)
 MUSTENTER, Wert
 DFHMDF [851](#)
 DFHMDI [862](#)
 DFHMSD [871](#)
 MUSTFILL, Wert
 DFHMDF [851](#)
 DFHMDI [862](#)
 DFHMSD [871](#)

N

Name, Argument, CICS-Befehlsformat [2](#)
 NAME, Option
 WAIT EVENT, Befehl [631](#)
 WAIT EXTERNAL, Befehl [633](#)
 WAITCICS, Befehl [640](#)

NAMELENGTH, Option
 WEB READ FORMFIELD, Befehl [675](#)
 WEB READ HTTPHEADER, Befehl [677](#)
 WEB READ QUERYPARM, Befehl [679](#)
 WEB READNEXT FORMFIELD, Befehl [680](#)
 WEB READNEXT HTTPHEADER, Befehl [682](#)
 WEB READNEXT QUERYPARM, Befehl [684](#)
 WEB STARTBROWSE FORMFIELD, Befehl [721](#)
 WEB STARTBROWSE QUERYPARM, Befehl [723](#)
 WEB WRITE HTTPHEADER, Befehl [726](#)

NATLANG, Option
 SIGNON TOKEN, Befehl [540](#)
 SIGNON, Befehl [535](#)

NATLANGINUSE, Option
 ASSIGN, Befehl [52](#)
 SIGNON TOKEN, Befehl [541](#)
 SIGNON, Befehl [535](#)

NETNAME, Option
 ASSIGN, Befehl [52](#)
 EXTRACT TCT, Befehl [188](#)

NETNAMEIDERR, Bedingung
 ALLOCATE (APPC), Befehl [37](#)

Neue Tasks, Daten übergeben [565](#)

NEWPASSWORD, Option
 CHANGE PASSWORD, Befehl [74](#)
 SIGNON, Befehl [536](#)

NEWPHRASE, Option
 CHANGE PHRASE, Befehl [72](#)
 SIGNON, Befehl [536](#)

NEWPHRASELEN, Option
 CHANGE PHRASE, Befehl [72](#)
 SIGNON, Befehl [536](#)

NEXT, Option
 READQ TS, Befehl [412](#)

NEXT, Wert
 DFHMDI [862](#)

NEXTTRANSID, Option
 ASSIGN, Befehl [52](#)

NLEOM, Option
 ROUTE, Befehl [475](#)
 SEND MAP, Befehl [510](#)
 SEND TEXT, Befehl [523](#)

NO, Wert
 DFHMDI [862](#)
 DFHMSD [871](#)

NOAUTOPAGE, Option
 SEND PAGE, Befehl [517](#)

NOCC, Option
 SPOOLOPEN OUTPUT, Befehl [556](#)

NOCHECK, Option
 DEFINE PROCESS, Befehl [113](#)
 START, Befehl [571](#)

NODATA, Option
 GET CONTAINER (BTS), Befehl [234](#)
 GET CONTAINER (CHANNEL), Befehl [238](#)
 GET64 CONTAINER, Befehl [262](#)

NODE, Option
 SPOOLOPEN OUTPUT, Befehl [556](#)

NODEIDERR, Bedingung
 SPOOLOPEN OUTPUT, Befehl [558](#)

NODUMP, Option
 ABEND, Befehl [27](#)

NOFLUSH, Option
 SEND MAP, Befehl [510](#)

NOHANDLE, Option
 Option [10](#)

NOJBUFSP, Bedingung
 WRITE JOURNALNAME, Befehl [737](#)

NONVAL, Bedingung
 ISSUE LOAD, Befehl [305](#)

NOPASSBKRD, Bedingung
 RECEIVE, Befehl (ohne z/OS Communications Server) [432](#)

NOPASSBKWR, Bedingung
 SEND (ohne z/OS Communications Server), Befehl [500](#)

NOQUEUE, Option
 ALLOCATE (APPC), Befehl [36](#)
 ALLOCATE (LUTYPE6.1), Befehl [38](#)
 ALLOCATE (MRO), Befehl [40](#)
 GDS ALLOCATE, Befehl [212](#)

NOQUIESCE, Option
 ISSUE PASS, Befehl [308](#)

NORM, Wert
 DFHMDF [851](#)

NOSPACE, Bedingung
 DUMP TRANSACTION, Befehl [158](#)
 REWRITE, Befehl [471](#)
 WRITE, Befehl [732](#)
 WRITEQ TD, Befehl [742](#)
 WRITEQ TS, Befehl [746](#)

NOSPOOL, Bedingung
 SPOOLCLOSE, Befehl [551](#)
 SPOOLOPEN INPUT, Befehl [553](#)
 SPOOLOPEN OUTPUT, Befehl [558](#)
 SPOOLREAD, Befehl [561](#)
 SPOOLWRITE, Befehl [564](#)

NOSTART, Bedingung
 ISSUE LOAD, Befehl [305](#)

NOSTG, Bedingung
 DUMP TRANSACTION, Befehl [158](#)
 GETMAIN, Befehl [248](#)
 GETMAIN64, Befehl [252](#)
 SPOOLCLOSE, Befehl [551](#)
 SPOOLOPEN INPUT, Befehl [553](#)
 SPOOLOPEN OUTPUT, Befehl [559](#)
 SPOOLREAD, Befehl [561](#)
 SPOOLWRITE, Befehl [564](#)

NOSUSPEND, Option
 ALLOCATE (APPC) [35](#)
 ALLOCATE (LUTYPE6.1), Befehl [38](#)
 Befehl REWIND DCOUNT [464](#)
 DEFINE COUNTER, Befehl [109](#)
 DEFINE DCOUNT, Befehl [109](#)
 DELETE [121](#)
 DELETE COUNTER, Befehl [134](#)
 DELETE DCOUNT, Befehl [134](#)
 ENQ, Befehl [166](#)
 FETCH CHILD, Befehl [194](#), [196](#)
 GET COUNTER, Befehl [240](#)
 GET DCOUNT, Befehl [240](#)
 GETMAIN, Befehl [247](#)
 GETMAIN64, Befehl [251](#)
 QUERY COUNTER, Befehl [364](#)
 READ, Befehl [379](#)
 READNEXT, Befehl [392](#)
 READPREV, Befehl [402](#)
 READQ TD, Befehl [408](#)
 REWIND COUNTER, Befehl [464](#)

NOSUSPEND, Option (*Forts.*)

REWRITE, Befehl [468](#)
UPDATE COUNTER, Befehl [615](#)
UPDATE DOUNTER, Befehl [615](#)
WRITE JOURNALNAME, Befehl [736](#)
WRITE, Befehl [729](#)
WRITEQ TS, Befehl [745](#)

NOTALLOC, Bedingung

CONNECT PROCESS, Befehl [83](#)
CONVERSE (Nicht-z/OS Communications Server), Befehl [103](#)
CONVERSE (z/OS Communications Server), Befehl [97](#)
EXTRACT ATTACH (LUTYPE6.1), Befehl [172](#)
EXTRACT ATTACH (MRO), Befehl [175](#)
EXTRACT ATTRIBUTES (APPC), Befehl [177](#)
EXTRACT ATTRIBUTES (MRO), Befehl [178](#)
EXTRACT LOGONMSG, Befehl [182](#)
EXTRACT PROCESS, Befehl [182](#)
EXTRACT TCT, Befehl [188](#)
FREE (APPC), Befehl [204](#)
FREE (LUTYPE6.1), Befehl [205](#)
FREE (MRO), Befehl [206](#)
FREE, Befehl [203](#)
ISSUE ABEND, Befehl [288](#)
ISSUE CONFIRMATION, Befehl [293](#)
ISSUE COPY (3270, logisch), Befehl [294](#)
ISSUE DISCONNECT (LUTYPE6.1), Befehl [296](#)
ISSUE ENDFILE, Befehl [299](#)
ISSUE ENDOUTPUT, Befehl [299](#)
ISSUE EODS, Befehl [300](#)
ISSUE ERASEAUP, Befehl [303](#)
ISSUE ERROR, Befehl [304](#)
ISSUE LOAD, Befehl [305](#)
ISSUE PASS, Befehl [308](#)
ISSUE PREPARE, Befehl [309](#)
ISSUE PRINT, Befehl [310](#)
ISSUE SIGNAL (APPC), Befehl [319](#)
ISSUE SIGNAL (LUTYPE6.1) Befehl [320](#)
POINT, Befehl [347](#)
RECEIVE (z/OS Communications Server), Befehl [424](#)
RECEIVE, Befehl (ohne z/OS Communications Server) [432](#)
SEND (ohne z/OS Communications Server), Befehl [500](#)
SEND (z/OS Communications Server), Befehl [496](#)
WAIT CONVID, Befehl [630](#)
WAIT SIGNAL, Befehl [638](#)
WAIT TERMINAL, Befehl [639](#)

NOTAUTH, Bedingung

ACQUIRE, Befehl [30](#)
CANCEL, Befehl [69](#)
CANCEL, Befehl (BTS) [70](#)
CHANGE PASSWORD, Befehl [73](#), [74](#)
DEFINE ACTIVITY, Befehl [106](#)
DEFINE PROCESS, Befehl [115](#)
DELETE, Befehl [121](#)
DELETEQ TD, Befehl [138](#)
DELETEQ TS, Befehl [140](#)
ENDBR, Befehl [161](#)
HANDLE ABEND, Befehl [265](#)
INQUIRE ACTIVITYID, Befehl [272](#)
INQUIRE CONTAINER, Befehl [273](#)
INQUIRE EVENT, Befehl [276](#)
INQUIRE PROCESS, Befehl [276](#)
INQUIRE TIMER, Befehl [278](#)

NOTAUTH, Bedingung (*Forts.*)

INVOKE APPLICATION, Befehl [281](#)
LINK ACQPROCESS, Befehl [333](#)
LINK ACTIVITY, Befehl [336](#)
LINK, Befehl [328](#)
LOAD, Befehl [339](#)
READ, Befehl [385](#)
READNEXT, Befehl [397](#)
READPREV, Befehl [407](#)
READQ TD, Befehl [410](#)
READQ TS, Befehl [413](#)
RELEASE, Befehl [440](#)
RESET ACQPROCESS, Befehl [447](#)
RESET ACTIVITY, Befehl [449](#)
RESETBR, Befehl [452](#)
REWRITE, Befehl [471](#)
RUN, Befehl [480](#)
SIGNON, Befehl [537](#), [542](#)
SPOOLOPEN INPUT, Befehl [553](#)
START ATTACH, Befehl [575](#)
START BREXIT, Befehl [576](#)
START TRANSID (CHANNEL), Befehl [579](#)
START, Befehl [565](#)
STARTBR, Befehl [588](#)
STARTBROWSE ACTIVITY, Befehl [591](#)
STARTBROWSE CONTAINER, Befehl [593](#)
STARTBROWSE EVENT, Befehl [595](#)
STARTBROWSE PROCESS, Befehl [596](#)
UNLOCK, Befehl [614](#)
VERIFY PASSWORD, Befehl [621](#)
VERIFY PHRASE, Befehl [624](#)
VERIFY TOKEN, Befehl [629](#)
WEB CONVERSE, Befehl [658](#)
WEB OPEN, Befehl [671](#)
WEB SEND, Befehl (Client) [719](#)
WRITE JOURNALNAME, Befehl [737](#)
WRITE, Befehl [733](#)
WRITEQ TD, Befehl [742](#)
WRITEQ TS, Befehl [747](#)
XCTL, Befehl [762](#)

NOTFND, Bedingung

CANCEL, Befehl [69](#)
DELETE COUNTER, Befehl [240](#), [464](#), [615](#)
DELETE, Befehl [121](#)
DOCUMENT RETRIEVE, Befehl [152](#)
QUERY SECURITY, Befehl [372](#)
READ, Befehl [385](#)
READNEXT, Befehl [397](#)
READPREV, Befehl [407](#)
RESETBR, Befehl [453](#)
REWRITE [470](#)
SOAPFAULT DELETE, Befehl [550](#)
SPOOLCLOSE, Befehl [551](#)
SPOOLOPEN INPUT, Befehl [554](#)
SPOOLOPEN OUTPUT, Befehl [559](#)
SPOOLREAD, Befehl [561](#)
STARTBR, Befehl [589](#)
WEB CONVERSE, Befehl [658](#)
WEB EXTRACT oder EXTRACT WEB, Befehl [193](#), [665](#)
WEB OPEN, Befehl [671](#)
WEB READ FORMFIELD, Befehl [676](#)
WEB READ HTTPHEADER, Befehl [678](#)
WEB READ QUERYPARM, Befehl [680](#)
WEB RECEIVE, Befehl (Server) [691](#)

NOTFND, Bedingung (*Forts.*)
 WEB RETRIEVE, Befehl [699](#)
 WEB SEND, Befehl (Client) [719](#)
 WEB SEND, Befehl (Server) [708](#)
 WEB STARTBROWSE FORMFIELD, Befehl [721](#)
 WEB STARTBROWSE HTTPHEADER, Befehl [722](#)
 WEB STARTBROWSE QUERYPARM, Befehl [724](#)

NOTOPEN, Bedingung
 DELETE, Befehl [121](#)
 DUMP TRANSACTION, Befehl [158](#)
 READ, Befehl [385](#)
 READQ TD, Befehl [410](#)
 SPOOLCLOSE, Befehl [551](#)
 SPOOLOPEN INPUT, Befehl [554](#)
 SPOOLOPEN OUTPUT, Befehl [559](#)
 SPOOLREAD, Befehl [562](#)
 SPOOLWRITE, Befehl [564](#)
 STARTBR, Befehl [589](#)
 UNLOCK, Befehl [614](#)
 WEB CLOSE, Befehl [644](#)
 WEB CONVERSE, Befehl [656](#)
 WEB ENDBROWSE HTTPHEADER, Befehl [660](#)
 WEB EXTRACT oder EXTRACT WEB, Befehl [193](#), [665](#)
 WEB READ HTTPHEADER, Befehl [678](#)
 WEB READNEXT HTTPHEADER, Befehl [683](#)
 WEB RECEIVE, Befehl (Client) [697](#)
 WEB SEND, Befehl (Client) [719](#)
 WEB STARTBROWSE HTTPHEADER, Befehl [722](#)
 WEB WRITE HTTPHEADER, Befehl [727](#)
 WRITE JOURNALNAME, Befehl [737](#)
 WRITE, Befehl [733](#)
 WRITEQ TD, Befehl [743](#)

NOTOPEN, Option
 WAIT JOURNALNAME, Befehl [636](#)

NOTTRUNCATE, Option
 CONVERSE (Nicht-z/OS Communications Server), Befehl [101](#)
 CONVERSE (z/OS Communications Server), Befehl [95](#)
 RECEIVE (z/OS Communications Server), Befehl [422](#)
 RECEIVE, Befehl (ohne z/OS Communications Server) [430](#)
 WEB CONVERSE, Befehl [653](#)
 WEB RECEIVE, Befehl (Client) [695](#)
 WEB RECEIVE, Befehl (Server) [688](#)

NOWAIT, Option
 ISSUE ADD, Befehl [291](#)
 ISSUE ERASE, Befehl [301](#)
 ISSUE REPLACE, Befehl [315](#)
 ISSUE SEND, Befehl [317](#)

NSCONTAINER, Option
 TRANSFORM XMLTODATA, Befehl [609](#)

NUM, Wert
 DFHMDF [851](#)

NUMBER, Wert
 DFHMDI [862](#)

NUMCIPHERS, Option
 WEB OPEN, Befehl [669](#)

NUMEVENTS, Option
 WAIT EXTERNAL, Befehl [633](#)
 WAITCICS, Befehl [640](#)

NUMITEMS, Option
 READQ TS, Befehl [412](#)
 WRITEQ TS, Befehl [745](#)

NUMREC, Option

NUMREC, Option (*Forts.*)
 DELETE, Befehl [121](#)
 ISSUE ADD, Befehl [291](#)
 ISSUE ERASE, Befehl [301](#)
 ISSUE REPLACE, Befehl [315](#)

NUMROUTES, Option
 WRITE OPERATOR, Befehl [739](#)

NUMSEGMENTS, Option
 DUMP TRANSACTION, Befehl [156](#)

NUMTAB, Option
 ASSIGN, Befehl [52](#)

O

OBFMT, Operand
 DFHMDI [862](#)
 DFHMDS [871](#)

OCCURS, Operand
 DFHMDF [851](#)

OFF, Wert
 DFHMDF [851](#)
 DFHMDI [862](#)
 DFHMDS [871](#)

OIDCARD, Option
 SIGNON, Befehl [536](#)

ON, Option
 DEFINE TIMER, Befehl [117](#)

OPCLASS, Option
 ASSIGN, Befehl [52](#)
 ROUTE, Befehl [475](#)

OPENERR, Bedingung
 DUMP TRANSACTION, Befehl [158](#)
 SPOOLOPEN INPUT, Befehl [554](#)
 SPOOLOPEN OUTPUT, Befehl [559](#)

OPERATION, Option
 INVOKE APPLICATION, Befehl [280](#)

OPERID, Option
 HANDLE AID, Befehl [265](#)

OPERKEYS, Option
 ASSIGN, Befehl [52](#)

OPERPURGE, Option
 SEND PAGE, Befehl [517](#)

OPID, Option
 ASSIGN, Befehl [53](#)

OPSECURITY, Option
 ASSIGN, Befehl [53](#)

Option NOHANDLE
 HANDLE AID überschreiben [11](#)

Optionen
 BMS [433](#), [436](#), [521](#)
 Länge [833](#)

OPTIONS(MAIN)
 in PL/I [12](#)

OR, Option
 DEFINE COMPOSITE EVENT, Befehl [108](#)

ORGABCODE, Option
 ASSIGN, Befehl [53](#)

ORGANIZATION, Option
 EXTRACT CERTIFICATE, Befehl [180](#)

ORGANIZATLEN, Option
 EXTRACT CERTIFICATE, Befehl [180](#)

ORGUNIT, Option
 EXTRACT CERTIFICATE, Befehl [180](#)

ORGUNITLEN, Option

ORGUNITLEN, Option (*Forts.*)
 EXTRACT CERTIFICATE, Befehl [180](#)
 OUTDESCR, Option
 SPOOLOPEN OUTPUT, Befehl [556](#)
 OUTDESCRERR, Bedingung
 SPOOLOPEN OUTPUT, Befehl [559](#)
 OUTLINE, Operand
 DFHMDf [851](#)
 DFHMDI [862](#)
 DFHMSD [871](#)
 OUTLINE, Option
 ASSIGN, Befehl [53](#)
 OUTPARTN, Option
 SEND CONTROL, Befehl [503](#)
 SEND MAP, Befehl [510](#)
 SEND TEXT NOEDIT, Befehl [530](#)
 SEND TEXT, Befehl [523](#)
 OVERFLOW, Bedingung
 SEND MAP, Befehl [512](#)
 OWNER, Option
 EXTRACT CERTIFICATE, Befehl [180](#)

P

PA1–PA3, Option
 HANDLE AID, Befehl [265](#)
 PAGE, Option
 SPOOLWRITE, Befehl [563](#)
 PAGENUM, Option
 ASSIGN, Befehl [53](#)
 PAGING, Option
 SEND CONTROL, Befehl [503](#)
 SEND MAP, Befehl [510](#)
 SEND TEXT MAPPED, Befehl [527](#)
 SEND TEXT NOEDIT, Befehl [530](#)
 SEND TEXT, Befehl [524](#)
 Partitionsdefinitions makro (DFHPDI) [850](#), [881](#)
 Partitionsgruppendefinitions makro (DFHPSD) [850](#), [882](#)
 PARTN, Operand
 DFHMDI [862](#)
 DFHMSD [871](#)
 PARTN, Option
 RECEIVE PARTN, Befehl [438](#)
 PARTNER, Option
 ALLOCATE (APPC), Befehl [36](#)
 CONNECT PROCESS, Befehl [82](#)
 GDS ALLOCATE, Befehl [212](#)
 GDS CONNECT PROCESS, Befehl [215](#)
 PARTNERIDERR, Bedingung
 ALLOCATE (APPC), Befehl [37](#)
 CONNECT PROCESS, Befehl [83](#)
 PARTNFAIL, Bedingung
 RECEIVE MAP, Befehl [435](#)
 PARTNPAGE, Option
 ASSIGN, Befehl [53](#)
 PARTNS, Option
 ASSIGN, Befehl [53](#)
 PARTNSET, Option
 ASSIGN, Befehl [53](#)
 PASSBK, Option
 RECEIVE, Befehl (ohne z/OS Communications Server) [430](#)
 SEND (ohne z/OS Communications Server), Befehl [499](#)
 PASSTICKET, Option
 REQUEST PASSTICKET [445](#)
 PASSWORD, Option
 CHANGE PASSWORD, Befehl [74](#)
 SIGNON, Befehl [536](#)
 VERIFY PASSWORD, Befehl [620](#)
 WEB CONVERSE, Befehl [650](#)
 WEB SEND, Befehl (Client) [715](#)
 PASSWORDLEN, Option
 WEB CONVERSE, Befehl [651](#)
 WEB SEND, Befehl (Client) [716](#)
 PATH, Option
 WEB CONVERSE, Befehl [651](#)
 WEB EXTRACT oder EXTRACT WEB, Befehl [190](#), [663](#)
 WEB PARSE URL, Befehl [673](#)
 WEB SEND, Befehl (Client) [716](#)
 PATHLENGTH, Option
 WEB CONVERSE, Befehl [651](#)
 WEB EXTRACT oder EXTRACT WEB, Befehl [190](#), [663](#)
 WEB PARSE URL, Befehl [673](#)
 WEB SEND, Befehl (Client) [716](#)
 PCT, Option
 DUMP TRANSACTION, Befehl [156](#)
 PF1–24, Option
 HANDLE AID, Befehl [265](#)
 PFXLENG, Option
 WRITE JOURNALNAME, Befehl [736](#)
 PGMIDERR, Bedingung
 HANDLE ABEND, Befehl [265](#)
 INVOKE APPLICATION, Befehl [281](#)
 LINK ACQPROCESS, Befehl [333](#)
 LINK ACTIVITY, Befehl [336](#)
 LINK, Befehl [328](#)
 LOAD, Befehl [339](#)
 RELEASE, Befehl [441](#)
 START BREXIT, Befehl [576](#)
 XCTL, Befehl [762](#)
 PHRASE, Option
 CHANGE PHRASE, Befehl [72](#)
 SIGNON, Befehl [536](#)
 VERIFY PHRASE, Befehl [624](#)
 PHRASELEN, Option
 CHANGE PHRASE, Befehl [72](#)
 SIGNON, Befehl [536](#)
 VERIFY PHRASE, Befehl [624](#)
 PICIN, Operand
 DFHMDf [851](#)
 PICOUT, Operand
 DFHMDf [851](#)
 Pipeline, logische Einheiten mit [417](#), [486](#)
 PIPLength, Option
 CONNECT PROCESS, Befehl [82](#)
 EXTRACT PROCESS, Befehl [182](#)
 GDS CONNECT PROCESS, Befehl [215](#)
 GDS EXTRACT PROCESS, Befehl [219](#)
 PIPLIST, Option
 CONNECT PROCESS, Befehl [82](#)
 EXTRACT PROCESS, Befehl [182](#)
 GDS CONNECT PROCESS, Befehl [215](#)
 GDS EXTRACT PROCESS, Befehl [219](#)
 PL/I-Sprache
 LENGTH, Option, Standard [7](#)
 PL/I, Programmiersprache
 Anweisung PROCEDURE [12](#)

PL/I, Programmiersprache (Forts.)
 Argumentwerte [6](#)
 konvertierter Code [12](#)
 STAE, Option [27](#)
 PLATFORM, Option
 INVOKE APPLICATION, Befehl [280](#)
 POINT, Befehl [347](#)
 POINT, Option
 MONITOR, Befehl [341](#)
 POOL, Option
 DEFINE COUNTER, Befehl [109](#)
 DEFINE DCOUNTER, Befehl [109](#)
 DELETE COUNTER, Befehl [134](#)
 DELETE DCOUNTER, Befehl [134](#)
 GET COUNTER, Befehl [240](#)
 GET DCOUNTER, Befehl [240](#)
 QUERY COUNTER, Befehl [364](#)
 QUERY DCOUNTER, Befehl [364](#)
 REWIND COUNTER, Befehl [464](#)
 REWIND DCOUNTER, Befehl [464](#)
 UPDATE COUNTER, Befehl [615](#)
 UPDATE DCOUNTER, Befehl [615](#)
 POP HANDLE, Befehl [348](#)
 PORTNUMBER, Option
 EXTRACT TCPIP, Befehl [186](#)
 WEB EXTRACT oder EXTRACT WEB, Befehl [190](#), [663](#)
 WEB OPEN, Befehl [669](#)
 WEB PARSE URL, Befehl [673](#)
 PORTNUMNU, Option
 EXTRACT TCPIP, Befehl [186](#)
 POS, Operand
 DFHMDI [851](#)
 POST, Befehl [348](#)
 Posten eines Steuerbereichs für Zeitgeberereignis [349](#)
 PPT, Option
 DUMP TRANSACTION, Befehl [156](#)
 PREDICATE, Option
 GETNEXT EVENT, Befehl [256](#)
 INQUIRE EVENT, Befehl [275](#)
 PREFIX, Option
 WRITE JOURNALNAME, Befehl [736](#)
 PRINCONVID, Option
 GDS ASSIGN, Befehl [214](#)
 PRINSYSID, Option
 ASSIGN, Befehl [53](#)
 GDS ASSIGN, Befehl [214](#)
 PRINT, Option
 ISSUE ABORT, Befehl [289](#)
 ISSUE END, Befehl [297](#)
 ISSUE SEND, Befehl [317](#)
 ISSUE WAIT, Befehl [321](#)
 SEND CONTROL, Befehl [504](#)
 SEND MAP MAPPINGDEV, Befehl [515](#)
 SEND MAP, Befehl [510](#)
 SEND TEXT NOEDIT, Befehl [530](#)
 SEND TEXT, Befehl [524](#)
 SPOOL OPEN OUTPUT, Befehl [557](#)
 PRINT, Wert
 DFHMDI [862](#)
 DFHMSD [871](#)
 Priorität einer Task ändern [75](#)
 PRIORITY, Option
 CHANGE TASK, Befehl [75](#)
 PRIVACY, Option (Forts.)
 EXTRACT TCPIP, Befehl [183](#)
 PROCESS, Option
 ACQUIRE, Befehl [29](#)
 ASSIGN, Befehl [54](#)
 BUILD ATTACH (LUTYPE6.1), Befehl [64](#)
 BUILD ATTACH (MRO), Befehl [66](#)
 DEFINE PROCESS, Befehl [114](#)
 DELETE CONTAINER (BTS), Befehl [132](#)
 EXTRACT ATTACH (LUTYPE6.1), Befehl [171](#)
 EXTRACT ATTACH (MRO), Befehl [174](#)
 GET CONTAINER (BTS), Befehl [234](#)
 GETNEXT PROCESS, Befehl [257](#)
 INQUIRE ACTIVITYID, Befehl [271](#)
 INQUIRE CONTAINER, Befehl [273](#)
 INQUIRE PROCESS, Befehl [276](#)
 PUT CONTAINER (BTS), Befehl [354](#)
 STARTBROWSE ACTIVITY, Befehl [591](#)
 STARTBROWSE CONTAINER, Befehl [592](#)
 PROCESSBUSY, Bedingung
 ACQUIRE, Befehl [30](#)
 CANCEL, Befehl (BTS) [70](#)
 DELETE CONTAINER (BTS), Befehl [133](#)
 GET CONTAINER (BTS), Befehl [235](#)
 LINK ACQPROCESS, Befehl [333](#)
 PUT CONTAINER (BTS), Befehl [355](#)
 RESET ACQPROCESS, Befehl [447](#)
 RUN, Befehl [480](#)
 PROCESSERR, Bedingung
 ACQUIRE, Befehl [30](#)
 CANCEL, Befehl (BTS) [70](#)
 DEFINE PROCESS, Befehl [115](#)
 GETNEXT PROCESS, Befehl [257](#)
 INQUIRE CONTAINER, Befehl [274](#)
 INQUIRE PROCESS, Befehl [276](#)
 LINK ACQPROCESS, Befehl [333](#)
 RESET ACQPROCESS, Befehl [447](#)
 RESUME, Befehl [454](#)
 RUN, Befehl [480](#)
 STARTBROWSE ACTIVITY, Befehl [591](#)
 STARTBROWSE CONTAINER, Befehl [593](#)
 STARTBROWSE PROCESS, Befehl [596](#)
 SUSPEND (BTS), Befehl [599](#)
 PROCESSTYPE, Option
 ACQUIRE, Befehl [29](#)
 ASSIGN, Befehl [54](#)
 DEFINE PROCESS, Befehl [114](#)
 INQUIRE ACTIVITYID, Befehl [271](#)
 INQUIRE CONTAINER, Befehl [273](#)
 INQUIRE PROCESS, Befehl [276](#)
 STARTBROWSE ACTIVITY, Befehl [591](#)
 STARTBROWSE CONTAINER, Befehl [592](#)
 STARTBROWSE PROCESS, Befehl [596](#)
 PROCLNGTH, Option
 CONNECT PROCESS, Befehl [82](#)
 EXTRACT PROCESS, Befehl [182](#)
 GDS CONNECT PROCESS, Befehl [215](#)
 GDS EXTRACT PROCESS, Befehl [219](#)
 PROCNAME, Option
 CONNECT PROCESS, Befehl [82](#)
 EXTRACT PROCESS, Befehl [182](#)
 GDS CONNECT PROCESS, Befehl [215](#)
 GDS EXTRACT PROCESS, Befehl [219](#)
 PROFILE, Option

PROFILE, Option *(Forts.)*
 ALLOCATE (APPC), Befehl [36](#)
 ALLOCATE (LUTYPE6.1), Befehl [38](#)
 ALLOCATE (MRO), Befehl [40](#)
 PROGRAM, Option
 ASSIGN, Befehl [54](#)
 DEFINE ACTIVITY, Befehl [105](#)
 DEFINE PROCESS, Befehl [114](#)
 DUMP TRANSACTION, Befehl [156](#)
 HANDLE ABEND, Befehl [264](#)
 INQUIRE ACTIVITYID, Befehl [271](#)
 ISSUE LOAD, Befehl [305](#)
 LINK, Befehl [325](#)
 LOAD, Befehl [338](#)
 RELEASE, Befehl [440](#)
 XCTL, Befehl [761](#)
 Programmsteuerung
 Befehle [22](#)
 geladenes Programm löschen [439](#)
 Programm, Tabelle oder Maske laden [337](#)
 Programmsteuerung zurückgeben [460](#)
 Übertragen der Programmsteuerung [759](#)
 PROT, Wert
 DFHMDf [851](#)
 PROTECT, Option
 START, Befehl [571](#)
 PS, Operand
 DFHMDf [851](#)
 DFHMDI [862](#)
 DFHMSD [871](#)
 PS, Option
 ASSIGN, Befehl [54](#)
 PSEUDOBIN, Option
 CONVERSE (Nicht-z/OS Communications Server), Befehl [101](#)
 RECEIVE, Befehl (ohne z/OS Communications Server) [430](#)
 SEND (ohne z/OS Communications Server), Befehl [499](#)
 psid, Wert
 DFHMDf [851](#)
 DFHMDI [862](#)
 DFHMSD [871](#)
 PUNCH, Option
 SPOOL OPEN OUTPUT, Befehl [557](#)
 PURGE MESSAGE, Befehl [352](#)
 PURGEABILITY, Option
 WAIT EXTERNAL, Befehl [633](#)
 WAITCICS, Befehl [640](#)
 PUSH HANDLE, Befehl [353](#)
 PUT CONTAINER (BTS), Befehl [353](#)
 PUT CONTAINER (CHANNEL), Befehl [355](#)
 PUT64 CONTAINER, Befehl [359](#)

Q

QBUSY, Bedingung
 READQ TD, Befehl [410](#)
 QIDERR, Bedingung
 DELETEQ TD, Befehl [138](#)
 DELETEQ TS, Befehl [140](#)
 QUERY SECURITY, Befehl [373](#)
 READQ TD, Befehl [410](#)
 READQ TS, Befehl [414](#)
 WRITEQ TD, Befehl [743](#)

QIDERR, Bedingung *(Forts.)*
 WRITEQ TS, Befehl [747](#)
 QNAME, Option
 ASSIGN, Befehl [54](#)
 DELETEQ TS, Befehl [139](#)
 READQ TS, Befehl [412](#)
 WRITEQ TS, Befehl [745](#)
 QUERY CHANNEL, Befehl [363](#)
 QUERY COUNTER, Befehl [364](#)
 QUERY DOUNTER, Befehl [364](#)
 QUERY SECURITY, Befehl [367](#)
 QUERYPARM, Option
 WEB READ QUERYPARM, Befehl [679](#)
 WEB READNEXT QUERYPARM, Befehl [683](#)
 WEB STARTBROWSE QUERYPARM, Befehl [723](#)
 QUERYSTRING, Option
 WEB EXTRACT oder EXTRACT WEB, Befehl [191](#), [663](#)
 WEB PARSE URL, Befehl [673](#)
 WEB SEND, Befehl [651](#)
 WEB SEND, Befehl (Client) [716](#)
 QUERYSTRLEN, Option
 WEB EXTRACT oder EXTRACT WEB, Befehl [191](#), [663](#)
 WEB PARSE URL, Befehl [673](#)
 WEB SEND, Befehl [651](#)
 WEB SEND, Befehl (Client) [716](#)
 QUEUE, Option
 BUILD ATTACH (LUTYPE6.1), Befehl [64](#)
 BUILD ATTACH (MRO), Befehl [66](#)
 DELETEQ TD, Befehl [138](#)
 DELETEQ TS, Befehl [139](#)
 EXTRACT ATTACH (LUTYPE6.1), Befehl [171](#)
 EXTRACT ATTACH (MRO), Befehl [174](#)
 READQ TD, Befehl [409](#)
 READQ TS, Befehl [412](#)
 RETRIEVE, Befehl [456](#)
 START, Befehl [571](#)
 WRITEQ TD, Befehl [741](#)
 WRITEQ TS, Befehl [745](#)
 QZERO, Bedingung
 READQ TD, Befehl [410](#)

R

RBA, Option
 DELETE, Befehl [121](#)
 READ, Befehl [379](#)
 READNEXT, Befehl [392](#)
 READPREV, Befehl [402](#)
 RESETBR, Befehl [450](#)
 STARTBR, Befehl [586](#)
 WRITE, Befehl [729](#)
 RDATT, Bedingung
 CONVERSE (Nicht-z/OS Communications Server), Befehl [103](#)
 RECEIVE MAP, Befehl [435](#)
 RECEIVE, Befehl (ohne z/OS Communications Server) [432](#)
 READ, Befehl [374](#)
 READ, Option
 QUERY SECURITY, Befehl [368](#)
 READNEXT, Befehl [387](#)
 READPREV, Befehl [398](#)
 READQ TS, Befehl [411](#)
 Reaktivieren eines Abbruchexits [263](#)

REALM, Option
 WEB EXTRACT oder EXTRACT WEB, Befehl [191](#), [664](#)
 REALMLen, Option
 WEB EXTRACT oder EXTRACT WEB, Befehl [191](#), [664](#)
 RECEIVE MAP MAPPINGDEV, Befehl [435](#)
 RECEIVE MAP, Befehl [432](#)
 RECEIVE PARTN, Befehl [437](#)
 RECEIVE-Befehl (logische 3270-Einheit) [417](#)
 RECEIVE, Befehl
 Daten aus Terminal oder logischer Einheit lesen [833](#)
 Eingabeoperation ohne Daten [837](#)
 RECEIVE, Befehl (2260) [425](#)
 RECEIVE, Befehl (2980) [426](#)
 RECEIVE, Befehl (3790- bzw. 3270-Anzeige) [428](#)
 RECEIVE, Befehl (APPC) [415](#)
 RECEIVE, Befehl (logische 3600- bzw. 3601-Einheit) [418](#)
 RECEIVE, Befehl (logische 3600- bzw. 3614-Einheit) [418](#)
 RECEIVE, Befehl (logische 3650-Einheit) [419](#)
 RECEIVE, Befehl (logische 3767-Einheit) [420](#)
 RECEIVE, Befehl (logische 3770-Einheit) [420](#)
 RECEIVE, Befehl (logische 3790-Einheit mit vollem Funktionsumfang oder Anfrage) [421](#)
 RECEIVE, Befehl (logische Einheit mit 3600-Pipeline) [417](#)
 RECEIVE, Befehl (LUTYPE2/LUTYPE3) [415](#)
 RECEIVE, Befehl (LUTYPE4) [416](#)
 RECEIVE, Befehl (LUTYPE6.1) [416](#)
 RECEIVE, Befehl (MRO) [425](#)
 RECEIVE, Befehl (ohne z/OS Communications Server) [424](#)
 RECEIVE, Befehl (Standardformat für z/OS Communications Server) [414](#)
 RECFM, Option
 BUILD ATTACH (LUTYPE6.1), Befehl [64](#)
 BUILD ATTACH (MRO), Befehl [67](#)
 EXTRACT ATTACH (LUTYPE6.1), Befehl [171](#)
 EXTRACT ATTACH (MRO), Befehl [174](#)
 RECORD, Option
 BIF DIGEST, Befehl [61](#)
 RECORDBUSY, Bedingung
 DELETE, Befehl [121](#)
 READ, Befehl [386](#)
 READNEXT, Befehl [397](#)
 READPREV, Befehl [407](#)
 REWRITE, Befehl [471](#)
 WRITE, Befehl [733](#)
 RECORDLEN, Option
 BIF DIGEST, Befehl [61](#)
 RECORDLENGTH, Option
 SPOOL OPEN OUTPUT, Befehl [557](#)
 REDUCE, Option
 GET COUNTER, Befehl [240](#)
 GET DCOUNTER, Befehl [240](#)
 Registerinhalt in Assemblersprache [13](#)
 Relative Byteadresse (RBA) [121](#)
 RELEASE, Befehl [439](#)
 RELEASE, Option
 SEND PAGE, Befehl [517](#)
 REMOVE SUBEVENT, Befehl [441](#)
 REPEATABLE, Option
 READ, Befehl [379](#)
 READNEXT, Befehl [392](#)
 READPREV, Befehl [403](#)
 REPLY, Option
 WRITE OPERATOR, Befehl [739](#)
 REPLYLENGTH, Option
 REPLYLENGTH, Option (*Forts.*)
 WRITE OPERATOR, Befehl [739](#)
 REQID, Option
 CANCEL, Befehl [68](#)
 DELAY, Befehl [118](#)
 ENDBR, Befehl [160](#)
 POST, Befehl [351](#)
 READNEXT, Befehl [392](#)
 READPREV, Befehl [403](#)
 RESETBR, Befehl [451](#)
 ROUTE, Befehl [475](#)
 SEND CONTROL, Befehl [504](#)
 SEND MAP, Befehl [511](#)
 SEND TEXT MAPPED, Befehl [527](#)
 SEND TEXT NOEDIT, Befehl [530](#)
 SEND TEXT, Befehl [524](#)
 START, Befehl [571](#)
 STARTBR, Befehl [586](#)
 WAIT JOURNALNAME, Befehl [636](#)
 WRITE JOURNALNAME, Befehl [736](#)
 REQUEST ENCRYPTPTKT, Befehl [442](#)
 REQUEST PASSTICKET, Befehl [444](#)
 REQUESTTYPE, Option
 WEB EXTRACT oder EXTRACT WEB, Befehl [191](#), [664](#)
 RESCLASS, Option
 QUERY SECURITY, Befehl [368](#)
 RESET ACQPROCESS, Befehl [446](#)
 RESET ACTIVITY, Befehl [447](#)
 RESET, Option
 HANDLE ABEND, Befehl [265](#)
 RESETBR, Befehl [449](#)
 RESID, Option
 QUERY SECURITY, Befehl [369](#)
 RESIDLENGTH, Option
 QUERY SECURITY, Befehl [370](#)
 RESOURCE, Option
 BUILD ATTACH (LUTYPE6.1) Befehl [64](#)
 BUILD ATTACH (MRO), Befehl [67](#)
 DEQ, Befehl [142](#)
 ENQ, Befehl [166](#)
 ENTER TRACENUM, Befehl [168](#)
 EXTRACT ATTACH (LUTYPE6.1), Befehl [172](#)
 EXTRACT ATTACH (MRO), Befehl [175](#)
 RESP
 Option [10](#)
 RESP2
 Bedingung EXPIRED in Nachrichten an Konsolenbediener [740](#)
 Bedingung INVREQ beim Befehl SIGNOFF (Sicherheitssteuerung) [533](#)
 Bedingung INVREQ beim Befehl SIGNON (Sicherheitssteuerung) [537](#), [541](#)
 Bedingung INVREQ beim Befehl WAIT EXTERNAL [634](#)
 Bedingung INVREQ beim Befehl WAITCICS [641](#)
 Bedingung INVREQ in Nachrichten an Konsolenbediener [740](#)
 Bedingung LENGERR in Nachrichten an Konsolenbediener [741](#)
 Bedingung NOTAUTH beim Befehl SIGNON (Sicherheitssteuerung) [538](#)
 Bedingung USERIDERR beim Befehl SIGNON (Sicherheitssteuerung) [538](#)
 Option [10](#)
 RESSEC, Option

- RESSEC, Option (*Forts.*)
 - ASSIGN, Befehl [54](#)
- Ressourcenplanung [140](#)
- RESTART, Option
 - ASSIGN, Befehl [54](#)
- RESTYPE, Option
 - QUERY SECURITY, Befehl [370](#)
- RESULT, Option
 - BIF DIGEST, Befehl [61](#)
 - integrierte Funktion [61](#)
- RESUME, Befehl [453](#)
- RESUNAVAIL, Bedingung
 - LINK, Befehl [329](#)
 - START TRANSID (CHANNEL), Befehl [579](#)
 - START, Befehl [565](#)
- RETAIN, Option
 - SEND PAGE, Befehl [517](#)
- RETCODE, Option
 - GDS ALLOCATE, Befehl [212](#)
 - GDS ASSIGN, Befehl [214](#)
 - GDS CONNECT PROCESS, Befehl [216](#)
 - GDS EXTRACT ATTRIBUTES, Befehl [217](#)
 - GDS EXTRACT PROCESS, Befehl [219](#)
 - GDS FREE, Befehl [220](#)
 - GDS ISSUE ABEND, Befehl [221](#)
 - GDS ISSUE CONFIRMATION, Befehl [223](#)
 - GDS ISSUE ERROR, Befehl [224](#)
 - GDS ISSUE PREPARE, Befehl [225](#)
 - GDS ISSUE SIGNAL, Befehl [227](#)
 - GDS RECEIVE, Befehl [228](#)
 - GDS SEND, Befehl [231](#)
 - GDS WAIT, Befehl [232](#)
- RETPAGE, Bedingung
 - SEND CONTROL, Befehl [505](#)
 - SEND MAP, Befehl [512](#)
 - SEND PAGE, Befehl [518](#)
 - SEND TEXT, Befehl [525](#)
- RETRIEVE REATTACH EVENT, Befehl [458](#)
- RETRIEVE SUBEVENT, Befehl [459](#)
- RETRIEVE, Befehl [455](#)
- RETURN, Befehl [460](#)
- RETURNPROG, Option
 - ASSIGN, Befehl [54](#)
- REVERSE, Wert
 - DFHMDF [851](#)
 - DFHMDI [862](#)
 - DFHMSD [871](#)
- REWIND COUNTER, Befehl [464](#)
- REWIND DOUNTER, Befehl [464](#)
- REWRITE, Befehl [467](#)
- REWRITE, Option
 - WRITEQ TS, Befehl [745](#)
- RIDFLD, Option
 - DELETE, Befehl [121](#)
 - ISSUE ADD, Befehl [291](#)
 - ISSUE ERASE, Befehl [301](#)
 - ISSUE NOTE, Befehl [306](#)
 - ISSUE REPLACE, Befehl [315](#)
 - READ, Befehl [379](#)
 - READNEXT, Befehl [392](#)
 - READPREV, Befehl [403](#)
 - RESETBR, Befehl [451](#)
 - STARTBR, Befehl [586](#)
 - WRITE, Befehl [729](#)

- RIGHT, Wert
 - DFHMDF [851](#)
 - DFHMDI [862](#)
- ROLLBACK, Option
 - SYNCPOINT ROLLBACK, Befehl [600](#)
- ROLLEDBACK, Bedingung
 - LINK, Befehl [330](#)
 - SYNCPOINT, Befehl [600](#)
- ROUTE, Befehl [472](#)
- ROUTECODES, Option
 - WRITE OPERATOR, Befehl [740](#)
- RPROCESS, Option
 - BUILD ATTACH (LUTYPE6.1) Befehl [64](#)
 - BUILD ATTACH (MRO), Befehl [67](#)
 - EXTRACT ATTACH (LUTYPE6.1), Befehl [172](#)
 - EXTRACT ATTACH (MRO), Befehl [175](#)
- RRESOURCE, Option
 - BUILD ATTACH (LUTYPE6.1) Befehl [65](#)
 - BUILD ATTACH (MRO), Befehl [67](#)
 - EXTRACT ATTACH (LUTYPE6.1), Befehl [172](#)
 - EXTRACT ATTACH (MRO), Befehl [175](#)
- RRN, Option
 - DELETE, Befehl [121](#)
 - ISSUE ADD, Befehl [291](#)
 - ISSUE ERASE, Befehl [301](#)
 - ISSUE NOTE, Befehl [306](#)
 - ISSUE REPLACE, Befehl [315](#)
 - READ, Befehl [380](#)
 - READNEXT, Befehl [393](#)
 - READPREV, Befehl [403](#)
 - RESETBR, Befehl [451](#)
 - STARTBR, Befehl [586](#)
 - WRITE, Befehl [730](#)
- RTEFAIL, Bedingung
 - ROUTE, Befehl [476](#)
- RTERMID, Option
 - RETRIEVE, Befehl [456](#)
 - START, Befehl [571](#)
- RTESOME, Bedingung
 - ROUTE, Befehl [476](#)
- RTRANSID, Option
 - RETRIEVE, Befehl [456](#)
 - START, Befehl [571](#)
- Rückgabe der Programmsteuerung [460](#)
- RUN, Befehl [476](#)

S

- SAA (Systems Application Architecture)
 - Kommunikation (CPI) [839](#)
 - Resource Recovery [839](#)
- SADDRLLENGTH, Option
 - EXTRACT TCPIP, Befehl [186](#)
- SAME, Wert
 - DFHMDI [862](#)
- SCHEME, Option
 - WEB EXTRACT oder EXTRACT WEB, Befehl [191](#), [664](#)
 - WEB OPEN, Befehl [669](#)
- SCHEMENAME, Option
 - WEB PARSE URL, Befehl [673](#)
- Schlüsselwortlänge [833](#)
- Schnittstellenprozessor DFHEAI [13](#)
- Schreiben von Daten
 - in Terminal oder logische Einheit [833](#)

Schreiben von Daten (*Forts.*)
 in Warteschlange für temporären Speicher [743](#)
 in Warteschlange mit transienten Daten [741](#)
Schreiben von Datensätzen in Dateien
 Dateisteuerung [727](#)
Schreiben von Datensätzen in Dateien (Datasets)
 Batch Data Interchange [290](#)
SCRNHT, Option
 ASSIGN, Befehl [55](#)
SCRNWD, Option
 ASSIGN, Befehl [55](#)
SCS (SNA-Zeichenfolge)
 Befehl SEND (z/OS Communications Server) [491](#)
 CONVERSE, Befehl [86](#)
 SEND, Befehl [485](#)
SCS-Drucker, logische Einheit mit 3790- [491](#)
SECONDS, Option
 DEFINE TIMER, Befehl [117](#)
 DELAY, Befehl [118](#)
 POST, Befehl [351](#)
 ROUTE, Befehl [475](#)
 START, Befehl [572](#)
SEGMENTLIST, Option
 DUMP TRANSACTION, Befehl [156](#)
Seitenwechsel durch Terminalbediener, Seitenwechselt-
ransaktion auslösen [516](#)
SELNERR, Bedingung
 ISSUE ABORT, Befehl [290](#)
 ISSUE ADD, Befehl [292](#)
 ISSUE END, Befehl [298](#)
 ISSUE ERASE, Befehl [302](#)
 ISSUE NOTE, Befehl [307](#)
 ISSUE QUERY, Befehl [311](#)
 ISSUE REPLACE, Befehl [315](#)
 ISSUE SEND, Befehl [318](#)
 ISSUE WAIT, Befehl [321](#)
SEND CONTROL, Befehl [500](#)
SEND MAP MAPPINGDEV, Befehl [513](#)
SEND MAP, Befehl [505](#)
SEND PAGE, Befehl [516](#)
SEND PARTNSET, Befehl [519](#)
SEND TEXT MAPPED, Befehl [526](#)
SEND TEXT NOEDIT, Befehl [528](#)
SEND TEXT, Befehl [519](#)
SEND, Befehl
 Daten in Terminal schreiben [833](#)
 SEND, Befehl (2260) [497](#)
 SEND, Befehl (2980) [497](#)
 SEND, Befehl (3790- bzw. 3270-Anzeige) [491](#)
 SEND, Befehl (3790- bzw. 3270-Drucker) [492](#)
 SEND, Befehl (APPC) [483](#)
 SEND, Befehl (logische 3270-Einheit) [485](#)
 SEND, Befehl (logische 3600- bzw. 3601-Einheit) [486](#)
 SEND, Befehl (logische 3600- bzw. 3614-Einheit) [487](#)
 SEND, Befehl (logische 3650- bzw. 3270-Einheit) [488](#)
 SEND, Befehl (logische 3650- bzw. 3653-Einheit) [489](#)
 SEND, Befehl (logische 3650- bzw. 3680-Einheit) [489](#)
 SEND, Befehl (logische 3767-Einheit) [490](#)
 SEND, Befehl (logische 3770-Einheit) [490](#)
 SEND, Befehl (logische 3790-Einheit mit vollem Funktions-
 umfang oder Anfrage) [491](#)
 SEND, Befehl (logische Einheit mit 3600-Pipeline) [486](#)
 SEND, Befehl (logische Einheit mit 3650-Interpreter) [488](#)
 SEND, Befehl (logische Einheit mit 3790-SCS-Drucker) [491](#)
 SEND, Befehl (LUTYPE2/LUTYPE3) [483](#)
 SEND, Befehl (LUTYPE4) [484](#)
 SEND, Befehl (LUTYPE6.1) [484](#)
 SEND, Befehl (MRO) [496](#)
 SEND, Befehl (SCS) [485](#)
 SEND, Befehl (Standardformat für z/OS Communications
 Server) [482](#)
 SEND, Befehl (Standardformat ohne z/OS Communications
 Server) [496](#)
 Senden eines asynchronen Interrupts [833](#)
 Senden von Daten an Ausgabeeinheit [316](#)
 Sequenzieller Abruf, Browsing
 Lesen von Datensätzen [374](#)
 SERIALNUM, Option
 EXTRACT CERTIFICATE, Befehl [180](#)
 SERIALNUMLEN, Option
 EXTRACT CERTIFICATE, Befehl [180](#)
 SERVADDRNU, Option
 EXTRACT TCPIP, Befehl [186](#)
 Server für benannten Zähler, Befehl REWIND [464](#)
 Server für benannten Zähler, Befehl UPDATE [615](#)
 SERVERADDR, Option
 EXTRACT TCPIP, Befehl [186](#)
 SERVERCONV, Option
 WEB RECEIVE (Server), Befehl [688](#)
 WEB SEND, Befehl (Server) [704](#)
 SERVERNAME, Option
 EXTRACT TCPIP, Befehl [186](#)
 SESSBUSY, Bedingung
 ALLOCATE (LUTYPE6.1), Befehl [39](#)
 SESSION, Option
 ALLOCATE (LUTYPE6.1), Befehl [38](#)
 CONNECT PROCESS, Befehl [82](#)
 CONVERSE (Nicht-z/OS Communications Server), Befehl
 101
 CONVERSE (z/OS Communications Server), Befehl [95](#)
 EXTRACT ATTACH (LUTYPE6.1), Befehl [172](#)
 EXTRACT ATTACH (MRO), Befehl [175](#)
 EXTRACT ATTRIBUTES (MRO), Befehl [177](#)
 FREE (LUTYPE6.1), Befehl [205](#)
 FREE (MRO), Befehl [205](#)
 ISSUE DISCONNECT (LUTYPE6.1), Befehl [296](#)
 ISSUE SIGNAL (LUTYPE6.1), Befehl [320](#)
 POINT, Befehl [347](#)
 RECEIVE (z/OS Communications Server), Befehl [423](#)
 RECEIVE, Befehl (ohne z/OS Communications Server)
 430
 SEND (ohne z/OS Communications Server), Befehl [499](#)
 SEND (z/OS Communications Server), Befehl [494](#)
 WAIT TERMINAL, Befehl [639](#)
 SESSIONERR, Bedingung
 ALLOCATE (LUTYPE6.1), Befehl [39](#)
 SESSTOKEN, Option
 WEB CLOSE, Befehl [643](#)
 WEB CONVERSE, Befehl [651](#)
 WEB ENDBROWSE HTTPHEADER, Befehl [659](#)
 WEB EXTRACT oder EXTRACT WEB, Befehl [192](#), [664](#)
 WEB OPEN, Befehl [669](#)
 WEB READ HTTPHEADER, Befehl [677](#)
 WEB READNEXT HTTPHEADER, Befehl [682](#)
 WEB RECEIVE, Befehl (Client) [695](#)
 WEB SEND, Befehl (Client) [716](#)
 WEB STARTBROWSE HTTPHEADER, Befehl [722](#)
 WEB WRITE HTTPHEADER, Befehl [726](#)

SET, Option

- ADDRESS SET, Befehl [34](#)
- CONVERSE (Nicht-z/OS Communications Server), Befehl [101](#)
- CONVERSE (z/OS Communications Server), Befehl [95](#)
- EXTRACT LOGONMSG, Befehl [181](#)
- GDS RECEIVE, Befehl [228](#)
- GET CONTAINER (BTS), Befehl [234](#)
- GET CONTAINER (CHANNEL), Befehl [238](#)
- GET64 CONTAINER, Befehl [262](#)
- GETMAIN, Befehl [247](#)
- GETMAIN64, Befehl [251](#)
- INQUIRE CONTAINER, Befehl [273](#)
- ISSUE RECEIVE, Befehl [312](#)
- LOAD, Befehl [338](#)
- POST, Befehl [351](#)
- READ, Befehl [380](#)
- READNEXT, Befehl [393](#)
- READPREV, Befehl [403](#)
- READQ TD, Befehl [409](#)
- READQ TS, Befehl [412](#)
- RECEIVE (z/OS Communications Server), Befehl [423](#)
- RECEIVE MAP MAPPINGDEV, Befehl [437](#)
- RECEIVE MAP, Befehl [434](#)
- RECEIVE PARTN, Befehl [439](#)
- RECEIVE, Befehl (ohne z/OS Communications Server) [430](#)
- RETRIEVE, Befehl [456](#)
- SEND CONTROL, Befehl [504](#)
- SEND MAP MAPPINGDEV, Befehl [515](#)
- SEND MAP, Befehl [511](#)
- SEND PAGE, Befehl [517](#)
- SEND TEXT, Befehl [524](#)
- WEB CONVERSE, Befehl [653](#)
- WEB READ FORMFIELD, Befehl [675](#)
- WEB READ QUERYPARM, Befehl [679](#)
- WEB RECEIVE, Befehl (Client) [695](#)
- WEB RECEIVE, Befehl (Server) [689](#)

SHARED, Option

- GETMAIN, Befehl [247](#)
- GETMAIN64, Befehl [251](#)

Sicherheit, Befehle [22](#)

Sicherstellen des Abschlusses einer Terminaloperation [638](#)

SIGDATA, Option

- ASSIGN, Befehl [55](#)

SIGNAL EVENT, Befehl [531](#)

SIGNAL, Bedingung

- CONVERSE (z/OS Communications Server), Befehl [97](#)
- ISSUE CONFIRMATION, Befehl [293](#)
- ISSUE DISCONNECT (Standard), Befehl [295](#)
- ISSUE ERROR, Befehl [304](#)
- RECEIVE (z/OS Communications Server), Befehl [424](#)
- SEND (z/OS Communications Server), Befehl [496](#)
- WAIT SIGNAL, Befehl [638](#)
- WAIT TERMINAL, Befehl [639](#)

SIGNOFF, Befehl [532](#)

SIGNON TOKEN, Befehl [538](#)

SIGNON, Befehl [533](#)

SIT, Option

- DUMP TRANSACTION, Befehl [156](#)

Sitzung übergeben [307](#)

SIZE, Operand

- DFHMDI [862](#)

SNAMELENGTH, Option

SNAMELENGTH, Option (*Forts.*)

- EXTRACT TCPIP, Befehl [186](#)

SOAPFAULT ADD, Befehl [543](#)

SOAPFAULT CREATE, Befehl [545](#)

SOAPFAULT DELETE, Befehl [549](#)

SOSI, Operand

- DFHMDI [851](#)
- DFHMDI [862](#)
- DFHMSD [871](#)

SOSI, Option

- ASSIGN, Befehl [55](#)

spalte, Wert

- DFHMDI [862](#)

Sparbuchsteuerung, 2980 [426](#)

SPCOMMAND

- RESID-Wert nicht gültig [372](#)

Speichersteuerung, Befehle [23](#)

SPOLBUSY, Bedingung

- SPOOLOPEN INPUT, Befehl [554](#)
- SPOOLOPEN OUTPUT, Befehl [559](#)

SPOLERR, Bedingung

- SPOOLOPEN INPUT, Befehl [554](#)
- SPOOLREAD, Befehl [562](#)
- SPOOLWRITE, Befehl [564](#)

Spool, Befehle [22](#)

SPOOLCLOSE, Befehl [550](#)

SPOOLCLOSE, impliziter Befehl [552](#)

SPOOLOPEN INPUT, Befehl [552](#)

SPOOLOPEN OUTPUT [555](#)

SPOOLREAD, Befehl [560](#)

SPOOLWRITE, Befehl [562](#)

Sprachencodes [831](#)

SRVRADDR6NU, Option

- EXTRACT TCPIP, Befehl [186](#)

SRVRIPFAMILY, Option

- EXTRACT TCPIP, Befehl [186](#)

SSLTYPE, Option

- EXTRACT TCPIP, Befehl [186](#)

STAE, Option, PL/I [27](#)

Standardliste für Attribute und Druckersteuerzeichen, BMS (DFHBMSCA) [841](#)

Stapelverarbeitung, logische Einheit 3770 [91](#), [420](#), [490](#)

START ATTACH, Befehl [575](#)

START CHANNEL, Befehl [579](#)

START, Befehl [565](#), [576](#)

STARTBR, Befehl [583](#)

STARTBROWSE ACTIVITY, Befehl [590](#)

STARTBROWSE CONTAINER (CHANNEL), Befehl [593](#)

STARTBROWSE CONTAINER, Befehl [591](#)

STARTBROWSE EVENT, Befehl [594](#)

STARTBROWSE PROCESS, Befehl [595](#)

STARTBROWSE TIMER, Befehl [596](#)

STARTCODE, Option

- ASSIGN, Befehl [55](#)

Starten einer Task [565](#)

STATE, Option

- ALLOCATE (APPC), Befehl [36](#)
- ALLOCATE (MRO), Befehl [40](#)
- CONNECT PROCESS, Befehl [82](#)
- CONVERSE (Nicht-z/OS Communications Server), Befehl [101](#)
- CONVERSE (z/OS Communications Server), Befehl [95](#)
- EXTRACT ATTRIBUTES (APPC), Befehl [176](#)
- EXTRACT ATTRIBUTES (MRO), Befehl [177](#)

STATE, Option (*Forts.*)
 EXTRACT CERTIFICATE, Befehl [180](#)
 FREE (APPC), Befehl [203](#)
 FREE (MRO), Befehl [205](#)
 GDS ALLOCATE, Befehl [212](#)
 GDS CONNECT PROCESS, Befehl [216](#)
 GDS EXTRACT ATTRIBUTES, Befehl [217](#)
 GDS FREE, Befehl [220](#)
 GDS ISSUE ABEND, Befehl [221](#)
 GDS ISSUE CONFIRMATION, Befehl [223](#)
 GDS ISSUE ERROR, Befehl [224](#)
 GDS ISSUE PREPARE, Befehl [225](#)
 GDS ISSUE SIGNAL, Befehl [227](#)
 GDS RECEIVE, Befehl [229](#)
 GDS SEND, Befehl [231](#)
 GDS WAIT, Befehl [232](#)
 ISSUE ABEND, Befehl [287](#)
 ISSUE CONFIRMATION, Befehl [292](#)
 ISSUE ERROR, Befehl [304](#)
 ISSUE PREPARE, Befehl [309](#)
 ISSUE SIGNAL (APPC), Befehl [318](#)
 RECEIVE (z/OS Communications Server), Befehl [423](#)
 RECEIVE, Befehl (ohne z/OS Communications Server) [431](#)
 SEND (ohne z/OS Communications Server), Befehl [499](#)
 SEND (z/OS Communications Server), Befehl [494](#)
 WAIT CONVID, Befehl [630](#)

STATELEN, Option
 EXTRACT CERTIFICATE, Befehl [180](#)

STATIONID, Option
 ASSIGN, Befehl [56](#)

STATUS, Option
 CHECK TIMER, Befehl [80](#)
 INQUIRE TIMER, Befehl [277](#)

STATUSCODE, Option
 WEB CONVERSE, Befehl [653](#)
 WEB RECEIVE, Befehl (Client) [695](#)
 WEB SEND, Befehl (Server) [705](#)

STATUSLEN, Option
 WEB CONVERSE, Befehl [653](#)
 WEB RECEIVE, Befehl (Client) [696](#)
 WEB SEND, Befehl (Server) [705](#)

STATUSTEXT, Option
 WEB CONVERSE, Befehl [653](#)
 WEB RECEIVE, Befehl (Client) [696](#)
 WEB SEND, Befehl (Server) [705](#)

STORAGE, Operand
 DFHMSD [871](#)

STORAGE, Option
 DUMP TRANSACTION, Befehl [156](#)

STRELERR, Bedingung
 SPOOLCLOSE, Befehl [552](#)
 SPOOLOPEN INPUT, Befehl [554](#)
 SPOOLOPEN OUTPUT, Befehl [559](#)
 SPOOLREAD, Befehl [562](#)
 SPOOLWRITE, Befehl [564](#)

STRFIELD, Option
 CONVERSE (Nicht-z/OS Communications Server), Befehl [101](#)
 CONVERSE (z/OS Communications Server), Befehl [96](#)
 SEND (ohne z/OS Communications Server), Befehl [499](#)
 SEND (z/OS Communications Server), Befehl [495](#)

STRINGFORMAT, Option
 FORMATTIME, Befehl [200](#)

STRINGZONE, Option
 FORMATTIME, Befehl [201](#)

Stub, Programm [13](#)

SUBADDR, Option
 ISSUE ABORT, Befehl [289](#)
 ISSUE END, Befehl [297](#)
 ISSUE SEND, Befehl [317](#)
 ISSUE WAIT, Befehl [321](#)

SUBEVENT, Option
 ADD SUBEVENT, Befehl [31](#)
 DEFINE COMPOSITE EVENT, Befehl [108](#)
 REMOVE SUBEVENT, Befehl [441](#)
 RETRIEVE SUBEVENT, Befehl [460](#)

Suchvorgang
 Ausgangspunkt zurücksetzen [449](#)
 nächsten Datensatz lesen [387](#)

SUFFIX, Operand
 DFHMSD [871](#)
 DFHPSD [882](#)

SUPPRESSED, Bedingung
 DUMP TRANSACTION, Befehl [159](#)
 GET COUNTER, Befehl [240](#)
 REWIND COUNTER, Befehl [464](#)
 UPDATE COUNTER, Befehl [615](#)
 WRITE, Befehl [734](#)

SUSPEND (BTS), Befehl [597](#)

SUSPEND, Befehl [597](#)

SUSPSTATUS, Option
 CHECK ACQPROCESS, Befehl [77](#)
 CHECK ACTIVITY, Befehl [79](#)
 INQUIRE ACTIVITYID, Befehl [271](#)

SYMBOL, Option
 DOCUMENT INSERT, Befehl [148](#)
 DOCUMENT SET, Befehl [153](#)

SYMBOLERR, Bedingung
 DOCUMENT SET, Befehl [154](#)

SYMBOLLIST, Option
 DOCUMENT SET, Befehl [144](#), [153](#)

Synchronisationsebenen
 Basisdialoge [219](#)

Synchronisationspunkt
 Befehle [23](#)
 erstellen [599](#)
 zurücksetzen auf [600](#)

Synchronisieren, Aktion
 Journalausgabe (Befehl WAIT JOURNALNAME) [635](#)
 Terminaleingabe/-ausgabe [833](#)

SYNCHRONOUS, Option
 RUN, Befehl [479](#)

SYNCLEVEL, Option
 CONNECT PROCESS, Befehl [83](#)
 EXTRACT PROCESS, Befehl [182](#)
 GDS CONNECT PROCESS, Befehl [216](#)
 GDS EXTRACT PROCESS, Befehl [219](#)

SYNCONRETURN, Option
 LINK, Befehl [326](#)

SYNCPPOINT ROLLBACK, Befehl [600](#)

SYNCPPOINT, Befehl [599](#)

Syntaxschreibweise [1](#)

SYSBUSY, Bedingung
 ALLOCATE (APPC), Befehl [37](#)
 ALLOCATE (LUTYPE6.1), Befehl [39](#)
 ALLOCATE (MRO), Befehl [40](#)

SYSID, Option

SYSID, Option (*Forts.*)

- ALLOCATE (APPC), Befehl [37](#)
- ALLOCATE (LUTYPE6.1), Befehl [38](#)
- ALLOCATE (MRO), Befehl [40](#)
- ASSIGN, Befehl [56](#)
- CANCEL, Befehl [68](#)
- DELETE, Befehl [121](#)
- DELETEQ TD, Befehl [138](#)
- DELETEQ TS, Befehl [139](#), [745](#)
- ENDBR, Befehl [160](#)
- EXTRACT TCT, Befehl [188](#)
- GDS ALLOCATE, Befehl [213](#)
- LINK, Befehl [326](#)
- READ, Befehl [380](#)
- READNEXT, Befehl [393](#)
- READPREV, Befehl [403](#)
- READQ TD, Befehl [409](#)
- READQ TS, Befehl [413](#)
- RESETBR, Befehl [451](#)
- REWRITE, Befehl [468](#)
- START TRANSID (CHANNEL), Befehl [579](#)
- START, Befehl [572](#)
- STARTBR, Befehl [586](#)
- UNLOCK, Befehl [613](#)
- WRITE, Befehl [730](#)
- WRITEQ TD, Befehl [741](#)

SYSIDERR, Bedingung

- ALLOCATE (APPC), Befehl [37](#)
- ALLOCATE (LUTYPE6.1), Befehl [39](#)
- ALLOCATE (MRO), Befehl [41](#)
- CANCEL, Befehl [69](#)
- DELETE, Befehl [121](#)
- DELETEQ TD, Befehl [139](#)
- DELETEQ TS, Befehl [140](#)
- ENDBR, Befehl [161](#)
- LINK, Befehl [330](#)
- READ, Befehl [386](#)
- READNEXT, Befehl [397](#)
- READPREV, Befehl [407](#)
- READQ TD, Befehl [410](#)
- READQ TS, Befehl [414](#)
- RESETBR, Befehl [453](#)
- REWRITE, Befehl [471](#)
- START TRANSID (CHANNEL), Befehl [579](#)
- START, Befehl [565](#)
- STARTBR, Befehl [589](#)
- UNLOCK, Befehl [614](#)
- WRITE, Befehl [734](#)
- WRITEQ TD, Befehl [743](#)
- WRITEQ TS, Befehl [747](#)

Systemname

- Definition [4–6](#), [8](#), [9](#)

Systemname, Argument, CICS-Befehlsformat [2](#)

T

Tabellen laden [337](#)

TABLES, Option

- DUMP TRANSACTION, Befehl [157](#)

Task

- Initialisierung [565](#)

Task, abnormale Beendigung [263](#)

TASK, Option

- DUMP TRANSACTION, Befehl [157](#)

Task, Verarbeitung verzögern [118](#)

TASKDATALOC, Option der Ressourcendefinition [33](#)

TASKPRIORITY, Option

- ASSIGN, Befehl [56](#)

Tasksteuerung, Befehle [23](#)

TCP/IP-Services [23](#)

TCPIPSERVICE, Option

- EXTRACT TCPIP, Befehl [187](#)

TCT, Option

- DUMP TRANSACTION, Befehl [157](#)

TCTUA, Option

- ADDRESS, Befehl [33](#)

TCTUALENG, Option

- ASSIGN, Befehl [56](#)

TELLERID, Option

- ASSIGN, Befehl [56](#)

TEMPLATE, Option

- DOCUMENT INSERT, Befehl [148](#)

Temporärer Speicher, Steuerbefehle [23](#)

TERM, Operand

- DFHMDSI [862](#)

- DFHMDS [871](#)

TERMCODE, Option

- ASSIGN, Befehl [56](#)

- Befehl ASSIGN [58](#)

TERMERR, Bedingung

- CONNECT PROCESS, Befehl [83](#)

- CONVERSE (Nicht-z/OS Communications Server), Befehl [103](#)

- CONVERSE (z/OS Communications Server), Befehl [97](#)

- ISSUE ABEND, Befehl [288](#)

- ISSUE CONFIRMATION, Befehl [293](#)

- ISSUE COPY (3270, logisch), Befehl [294](#)

- ISSUE DISCONNECT (LUTYPE6.1), Befehl [296](#)

- ISSUE DISCONNECT (Standard), Befehl [295](#)

- ISSUE EODS, Befehl [300](#)

- ISSUE ERASEAUP, Befehl [303](#)

- ISSUE ERROR, Befehl [304](#)

- ISSUE LOAD, Befehl [305](#)

- ISSUE PREPARE, Befehl [309](#)

- ISSUE PRINT, Befehl [310](#)

- ISSUE SIGNAL (APPC), Befehl [319](#)

- ISSUE SIGNAL (LUTYPE6.1) Befehl [320](#)

- LINK, Befehl [330](#)

- RECEIVE (z/OS Communications Server), Befehl [424](#)

- RECEIVE, Befehl (ohne z/OS Communications Server) [432](#)

- SEND (ohne z/OS Communications Server), Befehl [500](#)

- SEND (z/OS Communications Server), Befehl [496](#)

- WAIT SIGNAL, Befehl [638](#)

- WAIT TERMINAL, Befehl [639](#)

TERMID, Option

- EXTRACT TCT, Befehl [188](#)

- ISSUE COPY (3270, logisch), Befehl [294](#)

- START TRANSID (CHANNEL), Befehl [579](#)

- START, Befehl [572](#)

TERMIDERR, Bedingung

- START TRANSID (CHANNEL), Befehl [579](#)

- START, Befehl [565](#)

TERMINAL, Option

- DUMP TRANSACTION, Befehl [157](#)

- RECEIVE MAP, Befehl [434](#)

- SEND CONTROL, Befehl [504](#)

- SEND MAP, Befehl [511](#)

TERMINAL, Option (*Forts.*)
 SEND TEXT MAPPED, Befehl [527](#)
 SEND TEXT NOEDIT, Befehl [530](#)
 SEND TEXT, Befehl [524](#)
 Terminalmodellcodes [58](#)
 Terminalsteuerung
 Befehle [23](#)
 Terminalsysteem IBM 2980 für allgemeine Bankgeschäfte [426](#)
 Terminaltypcodes [58](#)
 Terminieren der Verwendung von Ressource durch Task [140](#), [164](#)
 TERMPRIORITY, Option
 ASSIGN, Befehl [56](#)
 TERMTHDACT [27](#)
 TEST EVENT, Befehl [601](#)
 TEXT, Option
 DOCUMENT INSERT, Befehl [149](#)
 WRITE OPERATOR, Befehl [740](#)
 TEXTKYBD, Option
 ASSIGN, Befehl [56](#)
 TEXTLENGTH, Option
 WRITE OPERATOR, Befehl [740](#)
 TEXTPRINT, Option
 ASSIGN, Befehl [56](#)
 Threadsichere Befehle [765](#)
 TIME, Option
 DELAY, Befehl [118](#)
 FORMATTIME, Befehl [201](#)
 POST, Befehl [351](#)
 ROUTE, Befehl [475](#)
 START, Befehl [565](#)
 TIMEDOUT, Bedingung
 WEB CONVERSE, Befehl [658](#)
 WEB EXTRACT oder EXTRACT WEB (Client), Befehl [193](#), [666](#)
 WEB OPEN, Befehl [671](#)
 WEB RECEIVE, Befehl (Client) [698](#)
 TIMEOUT, Option
 FETCH CHILD, Befehl [194](#), [196](#)
 WRITE OPERATOR, Befehl [740](#)
 TIMER, Option
 CHECK TIMER, Befehl [81](#)
 DEFINE TIMER, Befehl [117](#)
 DELETE TIMER, Befehl [137](#)
 FORCE TIMER, Befehl [197](#)
 GETNEXT EVENT, Befehl [256](#)
 INQUIRE EVENT, Befehl [275](#)
 INQUIRE TIMER, Befehl [277](#)
 TIMERERR, Bedingung
 CHECK TIMER, Befehl [81](#)
 DEFINE TIMER, Befehl [118](#)
 DELETE TIMER, Befehl [137](#)
 FORCE TIMER, Befehl [197](#)
 INQUIRE TIMER, Befehl [278](#)
 TIMESEP, Option
 FORMATTIME, Befehl [201](#)
 TIOAPFX, Operand
 DFHMDI [862](#)
 DFHMSD [871](#)
 TITLE, Option
 ROUTE, Befehl [476](#)
 TO, Option
 DOCUMENT INSERT, Befehl [149](#)
 TOACTIVITY, Option
 MOVE CONTAINER (BTS), Befehl [344](#)
 TOCHANNEL, Option
 MOVE CONTAINER (CHANNEL), Befehl [346](#)
 WEB CONVERSE, Befehl [653](#)
 WEB RECEIVE, Befehl (Client) [696](#)
 WEB RECEIVE, Befehl (Server) [689](#)
 TOCONTAINER, Option
 WEB CONVERSE, Befehl [654](#)
 WEB RECEIVE, Befehl (Client) [696](#)
 WEB RECEIVE, Befehl (Server) [690](#)
 TOLENGTH, Option
 CONVERSE (Nicht-z/OS Communications Server), Befehl [102](#)
 CONVERSE (z/OS Communications Server), Befehl [96](#)
 SPOOLREAD, Befehl [560](#)
 Vollwortalternative für TOLENGTH [833](#)
 TOKEN, Option
 DELETE, Befehl [121](#)
 READ, Befehl [380](#)
 READNEXT [393](#)
 READPREV, Befehl [404](#)
 REWRITE, Befehl [468](#)
 SIGNON TOKEN, Befehl [541](#)
 SPOOLCLOSE, Befehl [551](#)
 SPOOLOPEN INPUT, Befehl [552](#)
 SPOOLOPEN OUTPUT, Befehl [557](#)
 SPOOLREAD, Befehl [560](#)
 SPOOLWRITE, Befehl [563](#)
 UNLOCK, Befehl [613](#)
 VERIFY TOKEN, Befehl [627](#)
 TOKENERR, Bedingung
 ENDBROWSE ACTIVITY, Befehl [162](#)
 ENDBROWSE CONTAINER (CHANNEL), Befehl [163](#)
 ENDBROWSE CONTAINER, Befehl [162](#)
 ENDBROWSE EVENT, Befehl [163](#)
 ENDBROWSE PROCESS, Befehl [164](#)
 GETNEXT ACTIVITY, Befehl [253](#)
 GETNEXT CONTAINER (CHANNEL), Befehl [255](#)
 GETNEXT CONTAINER, Befehl [254](#)
 GETNEXT EVENT, Befehl [256](#)
 GETNEXT PROCESS, Befehl [257](#)
 WEB CONVERSE, Befehl [658](#)
 WEB SEND, Befehl (Client) [719](#)
 TOKENLEN, Option
 SIGNON TOKEN, Befehl [541](#)
 VERIFY TOKEN, Befehl [627](#)
 TOKENTYPE, Option
 SIGNON TOKEN, Befehl [541](#)
 VERIFY TOKEN, Befehl [627](#)
 TOLENGTH, Option
 CONVERSE (Nicht-z/OS Communications Server), Befehl [102](#)
 CONVERSE (z/OS Communications Server), Befehl [96](#)
 Vollwortalternative für Länge (TOLENGTH) [833](#)
 WEB CONVERSE, Befehl [654](#)
 TOPROCESS, Option
 MOVE CONTAINER (BTS), Befehl [344](#)
 TRACENUM, Option
 ENTER TRACENUM, Befehl [169](#)
 TRAILER, Operand
 DFHMDI [862](#)
 TRAILER, Option
 SEND PAGE, Befehl [517](#)

TRAILER, Option (*Forts.*)
 SEND TEXT, Befehl [524](#)
 TRANPRIORITY, Option
 ASSIGN, Befehl [57](#)
 TRANSFORM DATATOXML, Befehl [603](#)
 TRANSFORM XMLTODATA, Befehl [608](#)
 TRANSID, Option
 CANCEL, Befehl [68](#)
 DEFINE ACTIVITY, Befehl [106](#)
 DEFINE PROCESS, Befehl [114](#)
 INQUIRE ACTIVITYID, Befehl [271](#)
 LINK, Befehl [326](#)
 RETURN, Befehl [463](#)
 RUN TRANSID, Befehl [482](#)
 SEND PAGE, Befehl [518](#)
 START ATTACH, Befehl [575](#)
 START BREXIT, Befehl [576](#)
 START TRANSID (CHANNEL), Befehl [579](#)
 START, Befehl [565](#)
 TRANSIDERR, Bedingung
 DEFINE ACTIVITY, Befehl [107](#)
 DEFINE PROCESS, Befehl [115](#)
 START ATTACH, Befehl [575](#)
 START BREXIT, Befehl [576](#)
 START TRANSID (CHANNEL), Befehl [579](#)
 START, Befehl [565](#)
 Transiente Daten, Befehle [26](#)
 Transiente Daten, Steuerung
 Daten in Warteschlange mit transienten Daten schreiben [741](#)
 partitionsinterne Warteschlange löschen [137](#)
 TRANSP, Operand
 DFHMDF [851](#)
 DFHMDI [862](#)
 DFHMSD [871](#)
 Trennen einer Wählleitung [833](#)
 TRIGGER, Option
 HANDLE AID, Befehl [265](#)
 TRIGGER, Wert
 DFHMDF [851](#)
 DFHMDI [862](#)
 DFHMSD [871](#)
 TRIGRAPH, Operand
 DFHMSD [871](#)
 TRT, Option
 DUMP TRANSACTION, Befehl [158](#)
 TSIOERR, Bedingung
 PURGE MESSAGE, Befehl [352](#)
 SEND CONTROL, Befehl [505](#)
 SEND MAP, Befehl [512](#)
 SEND PAGE, Befehl [518](#)
 SEND TEXT MAPPED, Befehl [527](#)
 SEND TEXT NOEDIT, Befehl [531](#)
 SEND TEXT, Befehl [526](#)
 TWA, Option
 ADDRESS, Befehl [33](#)
 TWALENG, Option
 ASSIGN, Befehl [57](#)
 Typcodes für Terminals [58](#)
 TYPE, Operand
 DFHMSD [871](#)
 TYPE, Option
 WEB RECEIVE, Befehl (Server) [690](#)
 TYPENAME, Option

TYPENAME, Option (*Forts.*)
 TRANSFORM XMLTODATA, Befehl [609](#)
 TYPENAMELEN, Option
 TRANSFORM XMLTODATA, Befehl [609](#)
 TYPENS, Option
 TRANSFORM XMLTODATA, Befehl [609](#)
 TYPENSLEN, Option
 TRANSFORM XMLTODATA, Befehl [609](#)

U

Über 32K große COMMAREAs (Kanäle)
 ASSIGN, Befehl [47](#)
 Befehl START CHANNEL [579](#)
 DELETE CHANNEL, Befehl [130](#)
 DELETE CONTAINER (CHANNEL), Befehl [133](#)
 GET CONTAINER (CHANNEL), Befehl [236](#)
 MOVE CONTAINER (CHANNEL), Befehl [345](#)
 Option CHANNEL des Befehls XCTL [759](#)
 PUT CONTAINER (CHANNEL), Befehl [355](#)
 QUERY CHANNEL, Befehl [363](#)
 Übergabe der Steuerung
 bei Empfang eines AID-Zeichens (Befehl HANDLE AID) [265](#)
 ohne Rückgabe (XCTL) [759](#)
 Übergeben einer Sitzung [307](#)
 Übergeben von Daten an neue Tasks [565](#)
 Übersetzungsprogramm für Befehlssprachen
 konvertierter Code [13](#)
 Übertragen der Programmsteuerung [759](#)
 Überwachung der Anwendungsleistung [340](#)
 Überwachung, Befehle [22](#)
 Uhrzeit anfordern [41](#)
 Umgebungsservices
 Befehle [21](#)
 UNATTEND, Option
 ASSIGN, Befehl [57](#)
 UNCOMMITTED
 READ, Befehl [380](#)
 UNCOMMITTED, Option
 READNEXT [393](#)
 READPREV, Befehl [404](#)
 UNDERLINE, Wert
 DFHMDF [851](#)
 DFHMDI [862](#)
 DFHMSD [871](#)
 UNEXPIN, Bedingung
 ISSUE ABORT, Befehl [290](#)
 ISSUE ADD, Befehl [292](#)
 ISSUE END, Befehl [298](#)
 ISSUE ERASE, Befehl [302](#)
 ISSUE NOTE, Befehl [307](#)
 ISSUE QUERY, Befehl [312](#)
 ISSUE RECEIVE, Befehl [313](#)
 ISSUE REPLACE, Befehl [315](#)
 ISSUE SEND, Befehl [318](#)
 ISSUE WAIT, Befehl [322](#)
 RECEIVE MAP, Befehl [435](#)
 UNLOCK, Befehl [611](#)
 UNPROT, Wert
 DFHMDF [851](#)
 UNTIL, Option
 DELAY, Befehl [118](#)
 UPDATE COUNTER, Befehl [615](#)

- UPDATE DOUNTER, Befehl [615](#)
- UPDATE, Option
 - QUERY SECURITY, Befehl [372](#)
 - READ, Befehl [381](#)
 - READNEXT [394](#)
 - READPREV, Befehl [404](#)
- URIMAP, Option
 - WEB EXTRACT oder EXTRACT WEB, Befehl [192](#), [664](#)
 - WEB OPEN, Befehl [651](#), [669](#), [716](#)
- URL, Option
 - WEB PARSE URL, Befehl [673](#)
- URLLENGTH, Option
 - WEB PARSE URL, Befehl [673](#)
- USERDATAKEY, Option
 - GETMAIN, Befehl [248](#)
 - GETMAIN64, Befehl [251](#)
- USEREXIT, Wert
 - DFHMD5 [851](#)
 - DFHMDI [862](#)
 - DFHMSD [871](#)
- USERID, Option
 - ASSIGN, Befehl [57](#)
 - CHANGE PASSWORD, Befehl [74](#)
 - CHANGE PHRASE, Befehl [72](#)
 - DEFINE ACTIVITY, Befehl [106](#)
 - DEFINE PROCESS, Befehl [114](#)
 - EXTRACT CERTIFICATE, Befehl [181](#)
 - INQUIRE ACTIVITYID, Befehl [271](#)
 - SIGNON, Befehl [536](#)
 - SPOOL OPEN INPUT, Befehl [553](#)
 - SPOOL OPEN OUTPUT, Befehl [557](#)
 - START BREXIT, Befehl [576](#)
 - START TRANSID (CHANNEL), Befehl [579](#)
 - START, Befehl [565](#)
 - VERIFY PASSWORD, Befehl [620](#)
 - VERIFY PHRASE, Befehl [624](#)
- USERIDERR, Bedingung
 - CHANGE PASSWORD, Befehl [73](#), [75](#)
 - SIGNON, Befehl [538](#)
 - START BREXIT, Befehl [576](#)
 - START TRANSID (CHANNEL), Befehl [579](#)
 - START, Befehl [565](#)
 - VERIFY PASSWORD, Befehl [621](#)
 - VERIFY PHRASE, Befehl [625](#)
- USERNAME, Option
 - ASSIGN, Befehl [57](#)
 - WEB CONVERSE, Befehl [651](#)
 - WEB SEND, Befehl (Client) [716](#)
- USERSAMELEN, Option
 - WEB CONVERSE, Befehl [651](#)
 - WEB SEND, Befehl (Client) [717](#)
- USERPRIORITY, Option
 - ASSIGN, Befehl [57](#)
- USING, Option
 - ADDRESS SET, Befehl [34](#)
- UTC [201](#)

V

- VALIDATION, Option
 - ASSIGN, Befehl [57](#)
- VALIDN, Operand
 - DFHMD5 [851](#)
 - DFHMDI [862](#)

- VALIDN, Operand (Forts.)
 - DFHMSD [871](#)
- VALUE, Option
 - DEFINE COUNTER, Befehl [109](#)
 - DEFINE DOUNTER, Befehl [109](#)
 - DOCUMENT SET, Befehl [153](#)
 - GET COUNTER, Befehl [240](#)
 - GET DOUNTER, Befehl [240](#)
 - QUERY COUNTER, Befehl [364](#)
 - QUERY DOUNTER, Befehl [364](#)
 - UPDATE COUNTER, Befehl [615](#)
 - UPDATE DOUNTER, Befehl [615](#)
 - WEB READ FORMFIELD, Befehl [675](#)
 - WEB READ HTTPHEADER, Befehl [677](#)
 - WEB READ QUERYPARM, Befehl [679](#)
 - WEB READNEXT FORMFIELD, Befehl [680](#)
 - WEB READNEXT HTTPHEADER, Befehl [682](#)
 - WEB READNEXT QUERYPARM, Befehl [684](#)
 - WEB WRITE HTTPHEADER, Befehl [726](#)
- VALUELENGTH, Option
 - WEB READ FORMFIELD, Befehl [675](#)
 - WEB READ HTTPHEADER, Befehl [677](#)
 - WEB READ QUERYPARM, Befehl [679](#)
 - WEB READNEXT FORMFIELD, Befehl [680](#)
 - WEB READNEXT HTTPHEADER, Befehl [682](#)
 - WEB READNEXT QUERYPARM, Befehl [684](#)
 - WEB WRITE HTTPHEADER, Befehl [726](#)
- Verarbeitung einer Task, Verzögerung steuern [118](#)
- Verarbeitungsstatus einer Aktivität
 - ACTIVE [271](#)
 - CANCELLING [271](#)
 - COMPLETE [271](#)
 - DORMANT [271](#)
 - INITIAL [271](#)
- VERIFY PASSWORD, Befehl [618](#)
- VERIFY PHRASE, Befehl [621](#)
- VERIFY TOKEN, Befehl [625](#)
- Verschiebbarer Ausdruck [7](#)
- VERSIONLEN, Option
 - WEB EXTRACT oder EXTRACT WEB, Befehl [192](#), [664](#)
- Verzögerung der Verarbeitung, Task [118](#)
- VIEWPOS, Operand
 - DFHPDI [881](#)
- VIEWSZE, Operand
 - DFHPDI [881](#)
- Voller Funktionsumfang, logische Einheit [3790](#) [92](#)
- Vollwortoption für Länge [833](#)
- VOLUME, Option
 - ISSUE ABORT, Befehl [289](#)
 - ISSUE ADD, Befehl [291](#)
 - ISSUE END, Befehl [297](#)
 - ISSUE ERASE, Befehl [301](#)
 - ISSUE NOTE, Befehl [306](#)
 - ISSUE QUERY, Befehl [311](#)
 - ISSUE REPLACE, Befehl [315](#)
 - ISSUE SEND, Befehl [317](#)
 - ISSUE WAIT, Befehl [321](#)
- VOLUMELENG, Option
 - ISSUE ABORT, Befehl [290](#)
 - ISSUE ADD, Befehl [291](#)
 - ISSUE END, Befehl [298](#)
 - ISSUE ERASE, Befehl [301](#)
 - ISSUE NOTE, Befehl [306](#)
 - ISSUE QUERY, Befehl [311](#)

VOLUMELENG, Option (*Forts.*)
 ISSUE REPLACE, Befehl [315](#)
 ISSUE SEND, Befehl [317](#)
 ISSUE WAIT, Befehl [321](#)
 VSAM WRITE MASSINSERT
 Bedingung DISABLED nicht möglich [613](#)
 Bedingung NOTOPEN nicht möglich [614](#)
 Operation beenden [612](#)
 VTAB, Operand
 DFHMSD [871](#)

W

Wählleitungstrennung [833](#)
 WAIT CONVID, Befehl (APPC) [629](#)
 WAIT EVENT, Befehl [630](#)
 WAIT EXTERNAL, Befehl [632](#)
 WAIT JOURNALNAME, Befehl [635](#)
 WAIT JOURNALNUM, Befehl [637](#)
 WAIT SIGNAL, Befehl [637](#)
 WAIT TERMINAL, Befehl
 allgemeine Informationen [833](#)
 WAIT, Option
 Befehl SEND [833](#)
 GDS SEND, Befehl [231](#)
 ISSUE COPY (3270, logisch), Befehl [294](#)
 ISSUE ERASEAUP, Befehl [302](#)
 RETRIEVE, Befehl [456](#)
 SEND (ohne z/OS Communications Server), Befehl [499](#)
 SEND (z/OS Communications Server), Befehl [495](#)
 SEND CONTROL, Befehl [504](#)
 SEND MAP, Befehl [511](#)
 SEND TEXT MAPPED, Befehl [527](#)
 SEND TEXT NOEDIT, Befehl [530](#)
 SEND TEXT, Befehl [525](#)
 Terminalsteuerung [833](#)
 WRITE JOURNALNAME, Befehl [736](#)
 WAITCICS, Befehl [639](#)
 Wartestatus
 auf Auftreten eines Ereignisses [630](#)
 Batch Data Interchange [320](#)
 Terminalsteuerungsoperation [833](#)
 WEB CLOSE, Befehl [642](#)
 WEB CONVERSE, Befehl [644](#)
 WEB ENDBROWSE FORMFIELD, Befehl [658](#)
 WEB ENDBROWSE HTTPHEADER, Befehl [659](#)
 WEB ENDBROWSE QUERYPARM, Befehl [660](#)
 WEB EXTRACT, Befehl [660](#)
 WEB OPEN, Befehl [666](#)
 WEB PARSE URL, Befehl [671](#)
 WEB READ FORMFIELD, Befehl [674](#)
 WEB READ HTTPHEADER, Befehl [676](#)
 WEB READ QUERYPARM, Befehl [678](#)
 WEB READNEXT FORMFIELD, Befehl [680](#)
 WEB READNEXT HTTPHEADER, Befehl [681](#)
 WEB READNEXT QUERYPARM, Befehl [683](#)
 WEB RECEIVE, Befehl (Client) [691](#)
 WEB RECEIVE, Befehl (Server) [684](#)
 WEB RETRIEVE, Befehl [698](#)
 WEB SEND, Befehl (Client) [708](#)
 WEB SEND, Befehl (Server) [699](#)
 WEB STARTBROWSE FORMFIELD, Befehl [720](#)
 WEB STARTBROWSE HTTPHEADER, Befehl [721](#)
 WEB STARTBROWSE QUERYPARM, Befehl [722](#)

WEB WRITE HTTPHEADER, Befehl [724](#)
 Web-Services, Befehle [26](#)
 Webunterstützung [26](#)
 WPMEDIA, Option
 ISSUE ABORT, Befehl [290](#)
 ISSUE END, Befehl [298](#)
 ISSUE SEND, Befehl [317](#)
 ISSUE WAIT, Befehl [321](#)
 WRAP, Option
 GET COUNTER, Befehl [240](#)
 GET DCOUNTER, Befehl [240](#)
 WRBRK, Bedingung
 CONVERSE (Nicht-z/OS Communications Server), Befehl [103](#)
 SEND (ohne z/OS Communications Server), Befehl [500](#)
 SEND CONTROL, Befehl [505](#)
 SEND MAP, Befehl [512](#)
 SEND PAGE, Befehl [518](#)
 SEND TEXT MAPPED, Befehl [528](#)
 SEND TEXT NOEDIT, Befehl [531](#)
 SEND TEXT, Befehl [526](#)
 WRITE JOURNALNAME, Befehl [734](#)
 WRITE JOURNALNUM, Befehl [738](#)
 WRITE OPERATOR, Befehl
 kritische Aktion [739](#)
 sofortige Aktion [739](#)
 spätere Aktion [739](#)
 WRITE, Befehl [727](#)
 WRITEQ TD, Befehl [741](#)
 WRITEQ TS, Befehl [743](#)
 WSACONTEXT BUILD, Befehl [747](#)
 WSACONTEXT DELETE, Befehl [752](#)
 WSACONTEXT GET, Befehl [752](#)
 WSAEPR CREATE, Befehl [756](#)

X

XCTL, Befehl [759](#)
 XINIT, Operand
 DFHMD5 [851](#)
 XMLCONTAINER, Option
 TRANSFORM XMLTODATA, Befehl [609](#)
 XMLTRANSFORM, Option
 TRANSFORM XMLTODATA, Befehl [609](#)
 XRBA, Option
 DELETE, Befehl [121](#)
 READ, Befehl [381](#)
 READNEXT, Befehl [394](#)
 READPREV, Befehl [404](#)
 RESETBR, Befehl [451](#)
 STARTBR, Befehl [587](#)
 WRITE, Befehl [730](#)
 XRF, generische Anwendungs-ID (APPLID) [45](#)

Y

YEAR, Option
 DEFINE TIMER, Befehl [117](#)
 FORMATTIME, Befehl [201](#)
 YES, Wert
 DFHMDI [862](#)
 DFHMSD [871](#)
 YYDDD, Option

YYDDD, Option (*Forts.*)
 FORMATIME, Befehl [201](#)
YYDDMM, Option
 FORMATIME, Befehl [201](#)
YYMMDD, Option
 FORMATIME, Befehl [201](#)
YYYYDDD, Option
 FORMATIME, Befehl [201](#)
YYYYDDMM, Option
 FORMATIME, Befehl [202](#)
YYYYMMDD, Option
 FORMATIME, Befehl [202](#)

Z

z/OS Communications Server [84](#)
z/OS Communications Server, Zugriff auf Anmeldedaten [181](#)
zahl, Wert
 DFHMDf [851](#)
Zeigerverweis, Argument, CICS-Befehlsformat [2](#)
zeigerverweis64
 Befehlsformat [2](#)
Zeigerwert, Argument, CICS-Befehlsformat [2](#)
zeigerwert64
 Befehlsformat [2](#)
zeile, Wert
 DFHMDI [862](#)
zeile,spalte, Wert
 DFHMDf [851](#)
Zeitgeberereignis, Steuerbereich [349](#)
Zeitplanung, Befehle [22](#)
ZERO, Wert
 DFHMDf [851](#)
Zugriff auf Systeminformationen
 ADDRESS SET, Befehl [34](#)
 ADDRESS, Befehl [32](#)
 CICS-Speicherbereiche [32](#), [34](#)
Zurücksetzen auf einen Synchronisationspunkt [600](#)
Zurücksetzen des Starts für einen Suchvorgang [449](#)

