

# External Call Interface CICS TS

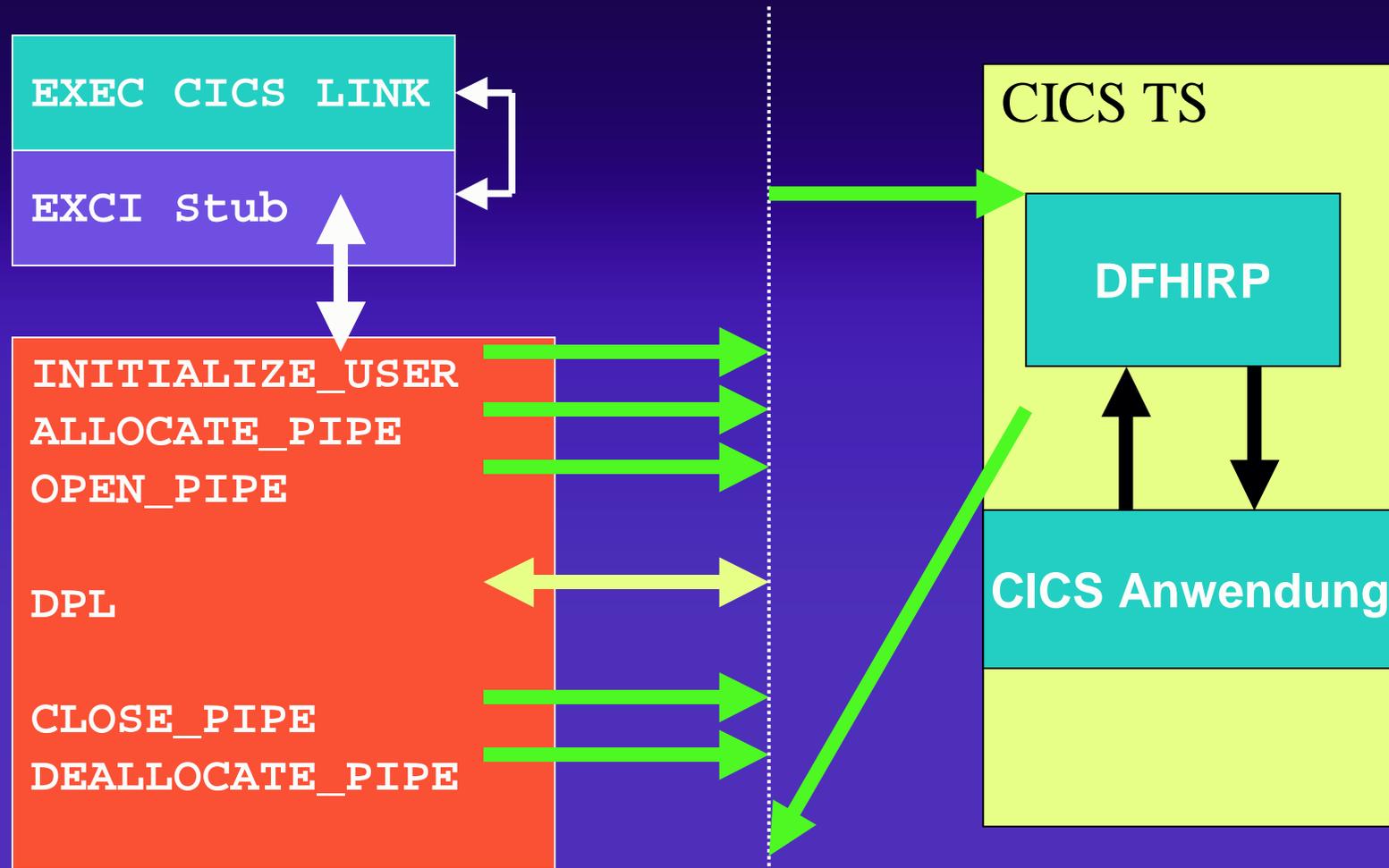
Nutzen von vorhandenen Funktionen  
Beispiele aus der Praxis

Heinz Peter Maassen  
Lattwein GmbH

# EXCI für CICS TS

- Installieren–Konfigurieren–Programmieren.
- OPEN-CLOSE von VSAM Dateien.
- VSAM Dateizugriffe über CICS.
- Programme im CICS mit Common Area aktivieren.
- Datenübergabe aus Batch.

# EXCI ~ Batch MRO zu CICS TS



# EXCI - Was ist das ?

- EXCI ermöglicht es Batch Anwendungen CICS Programme aufzurufen
  - ◆ Update von Ressourcen die innerhalb von CICS verwendet werden
    - ◆ Update von VSAM Files mit ShareOption 2
  - ◆ Ressourcen im CICS aktivieren / deaktivieren aus einer Batch Anwendung gestartet.
    - ◆ Dateien OPEN / CLOSE / UPDATE
    - ◆ Newcopy von Programmen
    - ◆ Enable / Disable von Transaktionen

# EXCI - Was ist das ?

- EXCI steht für **External Call Interface**
- Aufruf von CICS Programmen über **Distributed Program Link** aus Batch (wie MRO)
- Datenübergabe über die Commarea

# EXCI für CICS TS einrichten

- Im CEDA die Gruppe: DFH\$EXCI in eine Start Liste aufnehmen und installieren.
- Definiert werden:
  - ◆ Programm                    DFH\$AXCS
  - ◆ Transaktion                EXCI
  - ◆ Connections                EXCG, EXCS \*
  - ◆ Sessions                    EXCG, EXCS \*

\* Generic / Specific Connection/Session

# EXCI für CICS TS einrichten

- Generic / Specific Überlegungen:
- EXEC CICS LINK benutzt 'Generic' Connections und Sessions.
- Specific Connection werden verwendet, wenn ein **INITIALIZE\_USER** programmiert wird. Damit können mehrere Benutzer das gleiche Programm gleichzeitig benutzen – auch innerhalb eines Programms.

# EXCI – Group: DFH\$EXCI

```
Sitzung B - [24 x 80]
Datei Bearbeiten Sicht Kommunikation Aktionen Fenster Hilfe
EXP G (DFH$EXCI)
ENTER COMMANDS
NAME      TYPE      GROUP      DATE      TIME
EXCG      CONNECTION DFH$EXCI   02.021   09.42.03
EXCS      CONNECTION DFH$EXCI   02.021   09.42.02
DFH$AXCS  PROGRAM    DFH$EXCI *   02.021   09.42.01
EXCG      SESSIONS   DFH$EXCI *   02.021   09.42.04
EXCS      SESSIONS   DFH$EXCI *   02.021   09.42.03
EXCI      TRANSACTION DFH$EXCI *   02.021   09.42.01

RESULTS: 1 TO 6 OF 6
SYSID=CICT APPLID=CICSTEST
TIME: 13.07.43 DATE: 05.049
PF 1 HELP      3 END 4 TOP 5 BOT 6 CRSR 7 SBH 8 SFH 9 MSG 10 SB 11 SF 12 CNCL
MB b                                                    01/002
Verbindung zum Host über LU @L000005 aufgebaut.          \\Pserver\LEXMARK EIN \\Pserver\LEXMARK
```

# EXCI für CICS TS

- In der DFHSIT sollte **ISC** auf **YES** gesetzt werden.
- Aber auch: **IRCSTRT=YES**, dann kann **CEMT SET IRC OPEN** entfallen

# EXCI für CICS TS

- CEMT SET IRC OPEN (Wenn IRCSTRT=NO) !
- CEMT INQ Exci

STATUS: RESULTS

Exc( ECI.DFH\$AXCC.BG)

Tas(00000)

```
* $$ JOB JNM=TESTEXCI,CLASS=0,DISP=D
* $$ LST CLASS=V,DEST=(*,VSEMAINT)
// JOB EXCI AUFRUF
// ID USER=MAAS,PWD=PWDMAS (Muss angegeben sein) *
// EXEC PGM=DFH$AXCC,PARM='CICSTEST',OS390
/*
/&
* $$ EOJ
```

\* User Security Check ist Default weil: ATTACHCSEC(IDENTIFY)

# EXCI für CICS TS Batch

- Das Batch Programm wird mit CICS Preprocessor umgewandelt.
- Als Preprocessor Option muss EXCI gesetzt werden
- 
- `// EXEC DFHEAP1$,SIZE=500K`
- `*ASM XOPTS(EXCI)`
- 2 Möglichkeiten zum CICS Call

# EXCI .. Batch Single Call 1

- Aufbereiten der Daten in der COMMAREA um einen Server Call auszuführen.
- Aufruf über: **EXEC CICS LINK...**
- Bei Fehler: Abbruch der Verarbeitung
- Z.B. Antwort vom Server negativ: (z.B. Datei nicht gefunden, keine Verbindung oder andere ernsthafte Fehler)

# EXCI .. Batch Single Call 2

- Aufbereiten der Daten in der Commarea

```
MVC      TARGET_SYSTEM(8) , =CL8\CICSTEST\
```

```
TARGET_INFO      DS      0F  
TARGET_PROGRAM   DC      CL8'DFH$AXCS' *  
TARGET_TRANSID   DC      CL4'EXCI ,  
COMMAREA         DS      0F  
CALL             DS      F  
FILE             DS      CL8  
RIDFLD           DS      CL6  
RECORD           DS      CL80
```

\* = Programm Name im CICS TS

# EXCI .. Batch Single Call 3

\*=====

```
EXEC CICS LINK PROGRAM(TARGET_PROGRAM)
      TRANSID(TARGET_TRANSID)
      APPLID(TARGET_SYSTEM)
      COMMAREA(COMMAREA)
      LENGTH(EXEC_COM_LEN)
      DATALENGTH(EXEC_DAT_LEN)
      RETCODE(EXEC_RETAREA)
      SYNCONRETURN
```

\*=====

```
EXEC_COM_LEN      = Gesamt Länge der Common Area
EXEC_DAT_LEN      = Übergabe Länge ab Stelle 1
```

# EXCI .. Batch Single Call 4

- Die Commarea enthält einen Funktionsaufruf, sowie die Parameter die vom CICS Server Programm benötigt werden.
- Im Beispiel Programm DFH\$ACC sieht das wie folgt aus:
  - 1-4            Call = Funktion 1,2,3
  - 5-12          Dateiname
  - 13-18        KEY Feld
  - 19-98        Satz
- \* 0 1 2 3 4 5 6 7 8 9 A B C D E F 10 11 12 13
- \* ! CALL ! FILENAME ! RID ! RECORD

# EXCI .. Batch Single Call 5

- **Wenn EXEC\_RETAREA = F'0' ist wurde der Call ohne Fehler durchgeführt.**
- **Sonst sollte man alle möglichen Fehler analysieren und entsprechende Fehlermeldungen ausgeben.**

# EXCI .. Batch Multi CALL

- **Setup Init\_User call**
- 
- **Setup Allocate\_Pipe call**
- 
- **Setup Open\_Pipe call**
- **Setup DPL\_Request - Multipler Aufruf**
- **Setup Close\_PIPE**
- **Setup Deallocate\_Pipe**
- **Übersicht Response Codes - Return Area**

# EXCI .. Init\_User call

```
INITUSER DS      0H
          MVC     VERSION,=AL4(VERSION_1)           =AL4(1)
          MVC     EXCI_CALL,=AL4(INIT_USER)         =AL4(1)
*
          CALL    DFHXCIS,
              (VERSION,                             F
              CALL_RETAREA,                         CL12
              U_TOKEN,                              F
              EXCI_CALL,                            F
              APPLICATION),                          CL8`BATCHCLI`
          VL,
          MF=(E,PARM_LIST)
```

# EXCI .. Allocate\_Pipe call

```
ALOCPipe DS      0H
          MVC     EXCI_CALL,=AL4(ALLOCATE_PIPE)    =AL4(2)
          MVC     OPTIONS,=AL1(SPECIFIC_PIPE)      =AL1(0)
*
          CALL    DFHXCIS,
              (VERSION,                            F
              CALL_RETAREA,                        CL12
              U_TOKEN,                             F
              EXCI_CALL,                           AL4(2)
              P_TOKEN,                             F
              TARGET_SYSTEM,                        CL8`CICSTEST`
              OPTIONS),
          VL,
          MF=(E,PARM_LIST)
```

# EXCI .. Open\_Pipe call

```
OPENPIPE DS      0H
          MVC     EXCI_CALL,=AL4(OPEN_PIPE)           =AL4(3)
*
*
*
          CALL   DFHXCIS,
              (VERSION,                               F
              CALL_RETAREA,                           CL12
              U_TOKEN,                                 F
              EXCI_CALL,                               F
              P_TOKEN),                               F
              VL,
              MF=(E,PARM_LIST)
```

# EXCI .. DPL\_Request

```
DPLCALL DS 0H
        MVC EXCI_CALL,=AL4(DPL_REQUEST)    =AL4(6)
        MVC OPTIONS,=AL1(SYCONRETURN)     =XL1'80'
        SR  R2,R2                          keine Userid

*
        CALL DFHXCIS,
           (VERSION,CALL_RETAREA,U_TOKEN,P_TOKEN,
            EXCI_CALL,
            TARGET_PROGRAM,                CL8'DFH$AXCS'
            COMMAREA,
            COMM_LENGTH,                   COML_RECEIVE
            DATA_LENGTH,                  COML_SEND
            TARGET_TRANSID,                CL4'EXCI'
            0,R2),DPL_RETAREA,
            OPTIONS),
           VL,MF=(E,PARM_LIST)
```

# EXCI .. Close\_PIPE

```
CLOSPipe DS      0H
          MVC     EXCI_CALL,=AL4(CLOSE_PIPE)      =AL4(4)
*
          CALL   DFHXCIS,
              (VERSION,
              CALL_RETAREA,
              U_TOKEN,
              EXCI_CALL,
              P_TOKEN),
              VL,
              MF=(E,PARM_LIST)
*
```

# EXCI .. Deallocate\_Pipe

```
DEALPIPE DS      0H
          MVC     EXCI_CALL,=AL4(DEALLOCATE_PIPE) =AL4(5)
*
          CALL    DFHXCIS,
              (VERSION,
              CALL_RETAREA,
              U_TOKEN,
              EXCI_CALL,
              P_TOKEN),
          VL,
          MF=(E,PARM_LIST)
```

# EXCI .. Das Server Programm 1

- **Das Server Programm wird über den EXEX CICS LINK oder über den CALL DFHXCIS aufgerufen.**
- **Die Common Area muss als erstes analysiert werden.**
- **Dann erfolgt die gewohnte Ausführung von CICS Commands.**
- **Die Datenübergabe erfolgt ganz normal über die Common Area**

# EXCI .. Das Server Programm 2

- In diesem Programm sieht man keinen Unterschied zu einem anderen CICS Programm, dass über EXEC CICS LINK aufgerufen wird.
- Viele Möglichkeiten Batch und Online Verarbeitung zu synchronisieren.
- OPEN/CLOSE für Online Dateien aus Batch
- Im CICS eine Verarbeitung für Batch durchführen

# EXCI .. Das Server Programm 3

- Umgehung der Shareoption 4 und Updates im CICS durchführen.
- Batch gesteuerter Ablauf von Online Programmen
- Ein Beispiel aus dem Leben:
- OPEN / CLOSE aus Batch für VSAM Dateien.



# EXCI .. Beispiel

- Festlegung des Aufrufs:
- Da nur 1 Befehl im CICS durchgeführt werden soll, entscheiden wir uns für EXEC CICS LINK.
- Das Programm muss im **AMode 31** gelinkt werden, der **RMode** muss **24** sein.
- Das kann bei Reader, Punch, SYSLST und sequentiellen Dateien zu Problemen führen, da Amode temporär auf 24 geändert werden muss.

# EXCI .. Beispiel

## ▪ Batch

OPEN RDR,LST

Eingabe RDR  
Commarea aufbereiten  
EXEC CICS LINK

Close RDR,LST  
RETURN

## CICS

Common Area analysieren

OPEN – Check Open  
Wenn closed: Set File Open

CLOSE – Check CLOSE  
Wenn OPEN: Set File CLOSE

Common Area zurückgeben:  
RC+EC



# EXCI .. Beispiel Link

- Beachten: Link muss den STUB für EXCI haben, und zwar als letzten INCLUDE

```
// OPTION CATAL
```

```
  PHASE BACLIENT,*
```

```
  INCLUDE BACLIENT
```

```
  INCLUDE DFHXCSTB           muss 2. Include sein
```

```
/*
```

```
// EXEC LNKEDT,PARM='AMODE=31,RMODE=24'
```

**Beispiel Programme:** [www.lattwein.de](http://www.lattwein.de) dann Downloads

# EXCI .. Beispiel Job

```
// JOB ECI AUFRUF
```

```
// ID USER=MAAS,PWD=PWDMAS
```

```
// EXEC PGM=BACLIENT,OS390
```

```
CICSCPG2CPGWRK CLOSE
```

```
/*
```

```
// EXEC PROC=VSAMBKUP
```

```
// EXEC PGM=BACLIENT,OS390
```

```
CICSCPG2CPGWRK OPEN
```

```
/*
```

```
* CICSCPG2=APPLID, CPGWRK=VSAM Datei
```

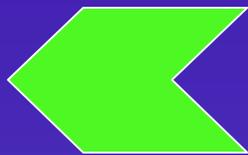
# EXCI .. Fazit

- ✓ Tolle Funktion und einfach zu realisieren.
- ✓ Aus Batch Abläufen CICS Programme aktivieren- oder Starten von Transaktionen.
- ✓ Open / Close für VSAM Dateien.
- ✓ Service für Dateien ändern READ-ONLY oder READ-WRITE.

# EXCI .. Fazit

- ✓ **Newcopy für Programme aus Batch.**
- ✓ **Transaktionen ENABLEN/ DISABLEN.**
- ✓ **Update von VSAM Dateien im CICS, wenn Datenvolumen nicht zu gross ist.**
- ✓ **Das sind nur Beispiele, viele andere ergeben sich in der Praxis.**

# EXCI ... Fragen ?



[hp.maassen@lattwein.de](mailto:hp.maassen@lattwein.de)

