**Note**

Before using this information and the product it supports, be sure to read the information in "Notices," on page 599.

---

Second Edition (February 2006)

This edition applies to version 5, release 4, modification 0 of i5/OS (product number 5722-SS1) and to all subsequent releases and modifications until otherwise indicated in new editions. This version does not run on all reduced instruction set computer (RISC) models nor does it run on CICS models.

© Copyright International Business Machines Corporation 1998, 2006. All rights reserved.
US Government Users Restricted Rights – Use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
Clocking (CLOCK) .......... 39
Line speed (INESPEED) .... 40
LMI mode (LMIMODE) ...... 40
Polling interval (POLLITY) 41
Full inquiry interval (FULLINQTV) 41
Text ‘description’ (TEXT) 41
Recovery limits (CMNRCYLMT) 41
Authority (AUT) ......... 42
Examples ................ 42
Error messages .......... 43

Create NWS Configuration (CRTNWSCFG) .......... 45
Parameters ................ 45
Network server configuration (NWSCFG) .... 46
Configuration type (TYPE) ...... 46
IP security rules (IPSECRULE) .... 47
Initialize service processor (INZSP) .... 47
Enable unicast (UNBUCAST) .... 48
Service processor name (SPNAME) ... 48
SP internet address (SPINTNETA) .... 49
SP certificate identifier (SPCERTID) .... 49
Enclosure identifier (EID) ....... 50
SP configuration name (SPNWSCFG) .... 50
Remote system identifier (RTMSYSID) .... 51
Delivery method (DELIVERY) .... 51
CHAP authentication (CHAPAUT) .... 51
Boot device ID (BOOTDEVID) .... 53
Dynamic boot options (DYNBOOTOPT) .... 53
Remote interfaces (RMTIFC) .... 54
Text ‘description’ (TEXT) .... 56
Authority (AUT) ......... 56
Examples ................ 57
Error messages .......... 57

Create Network Server Desc (CRTNWSD) .......... 59
Parameters ................ 59
Network server description (NWSD) ...... 62
Resource name (RSRCNAME) .... 62
Network server type (TYPE) ...... 62
Storage path (STGPTH) ....... 63
Default IP security rule (DFTSECRULE) .... 64
Multi-path group (MLTPTHGRP) .... 65
Default storage path (DFTSTGPTH) .... 65
Removable media path (RMVMDPTH) .... 65
Activation timer (ACTTMIR) .... 65
Online at IPL (ONLINE) .... 66
Vary on wait (VRYWAIT) .... 66
Shutdown timeout (SHUTDTIMO) .... 66
Partition (PARTITION) .... 67
Partition number (PTNNBR) .... 67
Domain role (DMNRROLE) .... 67
Propagate domain users (PRPDNMUSR) .... 68
Language version (LANGVER) .... 68
Code page (CODEPAGE) .... 70
Server message queue (MSGQ) .... 71
Event log (EVTLOG) ......... 71
Communications message queue (CMNMSGQ) .... 72
Configuration file (CFGFILE) .... 72
Server storage space sizes (SVRSTGSIZE) .... 73
Server storage space ASP (SVRSTGASP) .... 73
Server storage ASP device (STGASPDEV) .... 74
TCP/IP port configuration (TCPPORTCFG) .... 74
TCP/IP route configuration (TCPROUTE) .... 76
TCP/IP local host name (TCPHOSTNAME) .... 77
TCP/IP local domain name (TCPDMNNAME) .... 77
TCP/IP name server system (TCPNAMSVR) .... 79
Ports (PORTS) ............ 79
Virtual Ethernet path (VRTEHPATH) .... 80
Restricted device resources (RSTDEVRES) .... 81
Network server configuration (NWSCFG) .... 81
Shutdown TCP port (SHUTDPORT) .... 82
Virtual Ethernet control port (VRTEHCTRL) .... 82
Synchronize date and time (SYNCTIME) .... 83
IPL source (IPLSRC) .... 83
IPL stream file (IPLSTREAM) .... 84
IPL parameters (IPLPARM) .... 84
Power control (PRWCNTL) .... 84
Authority (AUT) ......... 84
Text ‘description’ (TEXT) .... 85
Examples ................ 85
Error messages .......... 86

Create NWS Storage Space (CRTNWSSSTG) .......... 87
Parameters ................ 87
Network server storage space (NWSSSTG) .... 87
Size (NWSSIZE) ......... 87
From storage space (FROMNWSSSTG) .... 88
Format (FORMAT) ......... 88
Auxiliary storage pool ID (ASP) .... 89
ASP device (ASPDEV) .... 89
Cluster domain name (CLUDMN) .... 89
Cluster port configuration (CLUDPORTCFG) .... 90
Text ‘description’ (TEXT) .... 90
Examples ................ 90
Error messages .......... 91

Create Output Queue (CRTOUTQ) .......... 93
Parameters ................ 93
Output queue (OUTQ) .... 94
Maximum spooled file size (MAXPAGES) .... 95
Order of files on queue (SEQ) .... 95
Remote system (RMTSYS) .... 95
Remote printer queue (RMTPTQ) .... 96
Writers to autostart (AUTOSTRWR) .... 97
Queue for writer messages (MSGQ) .... 97
Connection type (CNNTYPE) .... 97
Destination type (DESTTYPE) .... 97
Transform SCS to ASCII (TRANSFORM) .... 98
Data transform program (USRDATATFM) .... 98
Manufacturer type and model (MFRPTMPDL) .... 99
Workstation customizing object (WSCST) .... 105
Image configuration (IMGCFG) .... 106
Internet address (INNTNADDR) .... 108
VM/MVS class (CLASS) .... 108
Forms Control Buffer (FCB) .... 109
Destination options (DESTOPT) .... 109
Print separator page (SEPPAGE) .... 109
Parameters (USRDFOPT) ........................................... 110
User-defined options (USRDFOPT) .... 110
User-defined object (USRDFOBJ) .... 110
User-defined driver program (USRDVPGM) .. 111
Spooled file ASP (SPFAPI) .......... 111
Text ‘description’ (TEXT) ........ 111
Display any file (DSPDTA) ........ 112
Job separators (JOBSRC) .......... 112
Operator controlled (OPRCTRL) .... 112
Data queue (DTAQ) .................. 113
Authority to check (AUTCHK) .... 113
Authority (AUT) ................... 114
Examples ........................................ 114
Error messages .............................. 114

Create Overlay (CROTOVL) ............. 117
Parameters ........................................ 117
Overlay (OVL) .............................. 117
Source file (FILE) ..................... 118
Source file member (MBR) ........ 118
Data type (DATATYPE) .......... 118
Text ‘description’ (TEXT) .... 118
Replace overlay (REPLACE) ...... 119
Authority (AUT) ................... 119
Examples ........................................ 119
Error messages .............................. 120

Create Page Definition (CRTPGDFN) 121
Parameters ........................................ 121
Page definition (PAGDFN) ........ 121
File (FILE) .............................. 122
Member (MBR) ............................ 122
Text ‘description’ (TEXT) .... 122
Replace page definition (REPLACE) .... 122
Authority (AUT) ................... 123
Examples ........................................ 123
Error messages .............................. 123

Create Page Segment (CRTPGSEG) 125
Parameters ........................................ 125
Page segment (PAGSEG) ........ 125
Source file (FILE) ..................... 126
Source file member (MBR) .... 126
Text ‘description’ (TEXT) .... 126
Replace page segment (REPLACE) .... 126
Authority (AUT) ................... 127
Examples ........................................ 127
Error messages .............................. 127

Create Print Descriptor Group (CRTPDGDG) 129
Parameters ........................................ 129
Print descriptor group (PDG) .... 129
Text ‘description’ (TEXT) .... 129
Authority (AUT) ................... 130
Examples ........................................ 130
Error messages .............................. 130

Create PEX Data (CRTPDGEA) 133
Parameters ........................................ 133
From collection (FROMMGTCOL) ... 134
To member (TOMBR) ................. 134
To library (TOLIB) .................. 134
Number of threads (NBTHD) .. 134
Replace data (RPDLTA) .......... 135
Text ‘description’ (TEXT) .... 135
Examples ........................................ 135
Error messages .............................. 135

Create Physical File (CRTPF) 137
Parameters ........................................ 137
File (FILE) .................................. 138
Source file (SRCPFLE) .......... 139
Source member (SRCMBR) .... 139
Record length (RCDLEN) ..... 140
Generation severity level (GENNLV) .. 140
Flagging severity level (FLAG) .. 140
File type (FILETYPE) .......... 141
Member (MBR) ...................... 141
User specified DBCS data (IGCDTA) .. 141
Text ‘description’ (TEXT) .... 142
Source listing options (OPTION) .. 142
System (SYSTEM) .................. 143
Expiration date for member (EXPDATE) .. 143
Maximum members (MAXMBR) .... 143
Access path size (ACCPPTHLSIZ) ... 144
Access path logical page size (PAGESIZE) .. 144
Access path maintenance (MAINT) ... 144
Access path recovery (RECOVER) .. 145
Force keyed access path (FRKACCP) .. 146
Member size (SIZE) ................. 146
Allocate storage (ALLOCATE) ... 147
Contiguous storage (CONTIG) .. 147
Preferred storage unit (UNIT) .. 147
Records to force a write (FRCRATIO) .. 148
Maximum file wait time (WAITFILE) .. 148
Maximum record wait time (WAITRC) ... 148
Share open data path (SHARE) ... 148
Max % deleted records allowed (DLTPCT) .. 149
Reuse deleted records (REUISED) .. 149
Sort sequence (SORTSEQ) ... 149
Language ID (LANGID) ........ 150
Coded character set ID (CCSID) .. 150
Allow update operation (ALWUPD) .. 151
Allow delete operation (ALWDLT) .. 151
Record format level check (LVLCHK) .. 151
Node group (NODGRP) ............. 151
Partitioning Key (PTNKEY) ... 152
Authority (AUT) ................... 152
Examples ........................................ 152
Error messages .............................. 153

Create Performance Data (CRTPFRD) 155
Parameters ........................................ 155
From collection (FROMMGTCOL) ... 156
To member (TOMBR) ................. 156
To library (TOLIB) .................. 156
Text ‘description’ (TEXT) .... 156
Categories to process (CGY) ... 157
<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Create Program (CRTPGM)</strong></td>
<td>161</td>
</tr>
<tr>
<td>Parameters</td>
<td>161</td>
</tr>
<tr>
<td>Program (PGM)</td>
<td>162</td>
</tr>
<tr>
<td>Module (MODULE)</td>
<td>162</td>
</tr>
<tr>
<td>Text 'description' (TEXT)</td>
<td>163</td>
</tr>
<tr>
<td>Program entry procedure module (ENTMOD)</td>
<td>163</td>
</tr>
<tr>
<td>Bind service program (BNDSRVPGM)</td>
<td>164</td>
</tr>
<tr>
<td>Binding directory (BNDDIR)</td>
<td>164</td>
</tr>
<tr>
<td>Activation group (ACTGRP)</td>
<td>165</td>
</tr>
<tr>
<td>Creation options (OPTION)</td>
<td>165</td>
</tr>
<tr>
<td>Listing detail (DETAIL)</td>
<td>166</td>
</tr>
<tr>
<td>Allow update (ALWUPD)</td>
<td>167</td>
</tr>
<tr>
<td>Allow *SRVPGM library update (ALWLUPUPD)</td>
<td>167</td>
</tr>
<tr>
<td>User profile (USRPRF)</td>
<td>167</td>
</tr>
<tr>
<td>Replace program (REPLACE)</td>
<td>167</td>
</tr>
<tr>
<td>Authority (AUT)</td>
<td>168</td>
</tr>
<tr>
<td>Target release (TGTRLS)</td>
<td>168</td>
</tr>
<tr>
<td>Allow reinitialization (ALWREINIT)</td>
<td>169</td>
</tr>
<tr>
<td>Storage model (STGMDL)</td>
<td>169</td>
</tr>
<tr>
<td>Interprocedural analysis (IPA)</td>
<td>169</td>
</tr>
<tr>
<td>IPA control file (IPACTLFILE)</td>
<td>169</td>
</tr>
<tr>
<td>Examples</td>
<td>170</td>
</tr>
<tr>
<td>Error messages</td>
<td>170</td>
</tr>
</tbody>
</table>

| **Create Panel Group (CRTNLGRP)** | 171 |
| Parameters | 171 |
| Panel group (PNLGRP) | 172 |
| Source file (SRCFILE) | 172 |
| Source member (SRCMBR) | 172 |
| Text 'description' (TEXT) | 172 |
| Source listing options (OPTION) | 173 |
| Include file (INCFILE) | 173 |
| Character identifier (CHRID) | 174 |
| Authority (AUT) | 175 |
| Replace (REPLACE) | 175 |
| Examples | 175 |
| Error messages | 176 |

| **Create Printer File (CRTPTF)** | 177 |
| Parameters | 177 |
| File (FILE) | 181 |
| Source file (SRCFILE) | 182 |
| Source member (SRCMBR) | 182 |
| Generation severity level (GENLVL) | 182 |
| Flagging severity level (FLAG) | 183 |
| Device (DEV) | 183 |
| Printer device type (DEVTYP) | 183 |
| User specified DBCS data (IGCDTA) | 184 |
| DBCS extension characters (IGCENXCHR) | 184 |
| Text 'description' (TEXT) | 184 |
| Source listing options (OPTION) | 185 |
| Page size (PAGESIZE) | 185 |
| Lines per inch (LPI) | 186 |
| Characters per inch (CPI) | 187 |
| Front margin (FRONTMGN) | 187 |
| Back margin (BACKMGN) | 188 |
| Overflow line number (OVRFLOW) | 188 |
| Fold records (FOLD) | 189 |
| Unprintable character action (RPLUNPRT) | 189 |
| Align page (ALIGN) | 190 |
| Control character (CTLCRCHAR) | 190 |
| Channel values (CHLVAL) | 191 |
| Fidelity (FIDELITY) | 191 |
| Print quality (PRRTQITY) | 192 |
| Form feed (FORMFEED) | 192 |
| Source drawer (DRAWER) | 193 |
| Output bin (OUTBIN) | 193 |
| Font identifier (FONT) | 193 |
| Character identifier (CHRID) | 194 |
| Decimal format (DECFORM) | 195 |
| Font character set (FNTCHRS) | 195 |
| Coded font (CDFNFT) | 196 |
| Table Reference Characters (TBLREFCHR) | 197 |
| Page definition (PAGDEF) | 197 |
| Form definition (FORMDF) | 198 |
| AFP Characters (AFPCCHAR) | 199 |
| Degree of page rotation (PAGRTR) | 199 |
| Pages per side (MULTIPR) | 200 |
| Reduce output (REDUCE) | 200 |
| Print text (PRITXT) | 200 |
| Hardware justification (JUSTIFY) | 201 |
| Print on both sides (DUPLEX) | 201 |
| Unit of measure (UOM) | 201 |
| Front side overlay (FRONTOVL) | 202 |
| Back side overlay (BACKOVL) | 202 |
| Convert line data (CVTLINEDT) | 202 |
| IPDS pass through (IPDSPATHR) | 204 |
| User resource library list (USRSLRLST) | 204 |
| Corner staple (CORNERSTL) | 205 |
| Edge stitch (EDGESTICH) | 206 |
| Saddle stitch (SADDLSTICH) | 207 |
| Font resolution for formatting (RFNTRES) | 208 |
| Defer write (DFRWRT) | 208 |
| Spool the data (SPOOL) | 209 |
| Spooled output queue (OUTQ) | 209 |
| Form type (FORMTYP) | 209 |
| Copies (COPIES) | 210 |
| Expiration date for file (EXPDTE) | 210 |
| Days until file expires (DAYS) | 210 |
| Page range to print (PAGERANGE) | 210 |
| Max spooled output records (MAXOCD) | 211 |
| File separators (FILESEP) | 211 |
| Spooled output schedule (SCHEDULE) | 211 |
| Hold spooled file (HOLD) | 212 |
| Save spooled file (SAVE) | 212 |
| Output priority (OUTQ) (OUTPTY) | 212 |
| User data (USRDATA) | 212 |
| Spool file owner (SPLOOWN) | 212 |
| User Defined Option (USRDFOPT) | 213 |
| User Defined Data (USRDFDNTA) | 213 |
| User Defined Object (USRDFNOBJ) | 214 |
| DBCS character rotation (IGCCHRR) | 215 |
| DBCS characters per inch (IGCPPI) | 215 |
| DBCS SO/ST spacing (IGCSOS) | 215 |
| DBCS coded font (IGCCDEFN) | 216 |
Maximum file wait time (WAITFILE) ........................................ 216
Share open data path (SHARE) .................................................. 217
Record format level check (LVLCHK) ....................................... 217
Authority (AUT) ........................................................................ 217
Replace file (REPLACE) ............................................................. 218
Examples ................................................................................. 218
Error messages ......................................................................... 218

Create Proxy Command (CRTPRXCMD) ......................................... 219
Parameters ............................................................................... 219
Command (CMD) ....................................................................... 219
Target command (TGTCMD) ....................................................... 220
Text ‘description’ (TEXT) ......................................................... 220
Authority (AUT) ....................................................................... 221
Replace command (REPLACE) .................................................... 221
Examples ................................................................................. 222
Error messages ......................................................................... 222

Create PSF Configuration (CRTPSFCFG) ........................................ 223
Parameters ............................................................................... 223
PSF configuration (PSFCFG) ...................................................... 225
User resource library list (USRRSCLIBL) .................................... 225
Device resource library list (DEVRSCLIBL) ................................. 226
IPDS pass through (IPDSPASTHR) ............................................. 226
Activate release timer (ACTRLSTMR) ........................................ 227
Release timer (RLSTMR) ............................................................ 228
Restart timer (RESTRTM) ........................................................... 228
APPc and TCP/IP retry count (RETRY) ...................................... 229
Delay between APPc retries (RETRYDLY) ................................. 229
Acknowledgment frequency (ACKFRQ) .................................... 229
Printer response timer (PRTRSPTMR) ....................................... 230
Generate PDF output (PDFGEN) ................................................. 230
PDF device emulation type (PDFDEVTYP) ................................. 230
PDF paper size drawer 1 (PDFPPRDWR1) ................................. 231
PDF paper size drawer 2 (PDFPPRDWR2) ................................. 231
Multiple PDF files (PDFMULT) ............................................... 232
PDF fonts inline (PDFNCFNT) .................................................. 232
PDF data queue (PDFDTAQ) .................................................... 233
PDF mail server name (PDFMAILSVR) ...................................... 233
Sender of electronic mail (PDFSENDER) ................................. 233
PDF administrator (PDFADMIN) ............................................... 234
PDF user program (PDFMAPPGM) .......................................... 234
PDF mapping object (PDFMAP) .............................................. 235
PDF output queue (PDFOUTQ) .................................................. 235
PDF directory (PDFDIR) ........................................................... 236
Save AFP data (AFPSAVE) ..................................................... 236
AFP output queue (AFPOUTQ) ................................................ 237
Text ‘description’ (TEXT) ........................................................ 237
Automatic session recovery (AUTOSSNRCY) ............................ 237
Blank page (BLANKPAGE) ...................................................... 238
Page size control (PAGSIZCTL) .................................................. 238
Resident fonts (RESFONT) ........................................................ 239
Resource retention (RSCRET) ..................................................... 239
Edge origin (EDGEORIENT) .................................................... 239
Use outline fonts (USEOUTLFNT) ............................................. 240
PSF defined option (PSFDFNOPT) .......................................... 240
Font substitution messages (FNTSUBMSG) .............................. 240
Capture host fonts at printer (FNTCAPTURE) .......................... 240
Font resolution for formatting (FNTRSL) .................................. 240
Font mapping table (FNTTBL) .................................................. 241
Cut sheet emulation mode (CSEMODE) ..................................... 242
Use DBCS simulation fonts (MAPIGCNT) ............................... 242
Replace (REPLACE) ................................................................. 242
Authority (AUT) ................................................................... 242
Examples ............................................................................... 243
Error messages ....................................................................... 243

Create Query Management Form (CRTQMFORM) .......................... 245
Parameters ............................................................................... 245
Query management report form (QMFORM) ............................ 245
Source file (SRCFILE) ............................................................. 246
Source member (SRCMBR) ....................................................... 246
Text ‘description’ (TEXT) ........................................................ 246
Authority (AUT) ................................................................... 247
Replace object (REPLACE) ....................................................... 247
Examples ............................................................................... 247
Error messages ....................................................................... 248

Create Query Management Query (CRTQMQRY) ............................ 249
Parameters ............................................................................... 249
Query management query (QMQRY) ......................................... 249
Source file (SRCFILE) ............................................................. 250
Source member (SRCMBR) ....................................................... 250
Text ‘description’ (TEXT) ........................................................ 250
Sort sequence (SRTSEQ) .......................................................... 251
Language ID (LANGID) ........................................................... 251
Authority (AUT) ................................................................... 252
Replace object (REPLACE) ....................................................... 252
Examples ............................................................................... 252
Error messages ....................................................................... 253

Create Q/A Database (CRTQSTDB) ............................................. 255
Parameters ............................................................................... 255
Q/A database (QSTDB) ............................................................ 255
Lib containing Q/A database (LIB) ........................................... 255
Examples ............................................................................... 256
Error messages ....................................................................... 256

Create Q/A Database Load (CRTQSTLOD) ..................................... 257
Parameters ............................................................................... 257
Q/A database (QSTDB) ............................................................ 257
Lib containing Q/A database (LIB) ........................................... 257
Examples ............................................................................... 258
Error messages ....................................................................... 258

Create S/36 Display File (CRTS36DSPF) ....................................... 259
Parameters ............................................................................... 259
Display file (DSPFILE) ............................................................ 260
Option (OPTION) .................................................................... 260
S/36 SFGR source member (SRCMBR) ...................................... 260
S/36 source file (SRCFILE) ..................................................... 261
Replace display file (REPLACE) ............................................... 261
Print SFGR listing (PRINT) ....................................................... 261

Contents vii
Maximum devices (MAXDEV) .......................... 262
Authority (AUT) .................................. 262
Generation option (GENOPT) ...................... 263
Check SFRG source syntax (SYNTAX) ............. 263
To DDS source file (TOFILE) ...................... 263
To DDS source member (TOMBR) ................... 264
Issue msg if error occurs (HALT) ................. 264
Defer write (DFRWRT) ............................ 264
Target Release (TGTRLS) .......................... 265
Examples ......................................... 265
Error messages .................................... 265

Create S/36 Menu (CRTS36MNU) ................. 267
Parameters ........................................ 267
Command text source member# (CMDTXTMBR) .... 268
Option text source file (OPTTXTMBR) ............ 268
Command text source file (CMDTXTSRC) ......... 268
Option text source file (OPTTXTSRC) ............ 269
Menu library (LOADLIB) (MNULIB) ............... 269
Replace menu (REPLACE) .......................... 269
Free form menu (FREEFORM) ...................... 270
Keep option text msg file (KEEP) ................. 270
DDS listing (DDSLIST) ............................ 270
Maximum devices (MAXDEV) ....................... 270
Authority (AUT) .................................. 270
To DDS source file (TOFILE) ...................... 271
To DDS source member (TOMBR) ................... 272
User specified DBCS data (IGCDTA) ............... 272
Target Release (TGTRLS) .......................... 272
Examples ......................................... 273
Error messages .................................... 273

Create S/36 Message File (CRTS36MSGF) ......... 275
Parameters ........................................ 276
S/36 message source member (SRCMBR) ........... 276
S/36 source file (SRCFILE) ....................... 276
Message file library (MSGLIB) .................... 277
Replace message file (REPLACE) .................. 277
Message identifier prefix (MSGPFX) .............. 277
Option (OPTION) .................................. 277
Allow # substitution fields (SUBST) .............. 278
Enforce S/36 restrictions (RESTRICT) ............ 278
Authority (AUT) .................................. 279
To CL source file (TOFILE) ....................... 279
To CL source member (TOMBR) .................... 280
Issue msg if error occurs (HALT) ................. 280
Examples ......................................... 280
Error messages .................................... 281

Create Save File (CRTSAVF) ....................... 283
Parameters ........................................ 283
Save file (FILE) ................................. 284
Text ‘description’ (TEXT) ......................... 284
Maximum records (MAXRCDS) ...................... 284
Auxiliary storage pool ID (ASP) ................... 284
Maximum file wait time (WAITFILE) .............. 285
Share open data path (SHARE) .................... 285
Authority (AUT) .................................. 285
Examples ......................................... 285
Error messages .................................... 286

Error messages .................................... 286

Create Subsystem Description (CRTSSBD) ....... 287
Parameters ........................................ 287
Subsystem description (SBSD) ..................... 288
Storage pools (POOLS) ............................ 288
Maximum jobs (MAXJOBS) ........................ 289
Text ‘description’ (TEXT) ........................ 290
Sign-on display file (SGNDISP) .................... 290
Subsystem library (SYSLIB) ....................... 290
Authority (AUT) .................................. 291
Examples ......................................... 291
Error messages .................................... 292

Create Search Index (CRTSCHIDX) .............. 293
Parameters ........................................ 293
Search index (SCHIDX) ............................ 293
Display title (TITLE) ............................. 294
Text ‘description’ (TEXT) ........................ 294
Character identifier (CHRID) ..................... 294
Authority (AUT) .................................. 295
Examples ......................................... 295
Error messages .................................... 295

Create Spelling Aid Dictionary (CRTSPADCT) ...... 297
Parameters ........................................ 298
Spelling aid dictionary (SPADCT) ............... 298
Source file (SRCFILE) ............................. 299
Source member (SRCMBR) .......................... 299
Text ‘description’ (TEXT) ........................ 299
Base dictionary (BASEDCT) ....................... 299
Verify dictionary (VFYDCT) ....................... 300
Language attribute (LANGATR) .................... 300
Stop word list language ID (SWLLANGID) ........ 301
Base stop word list (BASESWL) ................... 302
Source listing option (OPTION) ................. 302
Authority (AUT) .................................. 302
Replace dictionary (REPLACE) .................... 303
Examples ......................................... 303
Error messages .................................... 303

Create SQL Package (CRTSQLPKG) ............... 305
Parameters ........................................ 305
Program (PGM) ................................... 305
Relational database (RDB) ......................... 306
RDB user (USER) .................................. 306
RDB user password (PASSWORD) .................... 306
Default collection (DFTRDBCOL) .................. 306
Object type (OBJTYPE) ............................ 307
Module list (MODULE) ............................. 307
Text ‘description’ (TEXT) ......................... 307
Severity level (GENLVL) .......................... 308
Replace (REPLACE) ................................ 308
Print file (PRINTFILE) ............................ 308
Examples ......................................... 308
Error messages .................................... 309
Create Source Physical File
(CRTSRCPF) .......................... 311
Parameters .......................... 311
File (FILE) .......................... 312
Record length (RCDLEN) ............... 312
Member (MBR) ........................ 313
User specified DBCS data (IGCDTA) ..... 313
Text ‘description’ (TEXT) ............... 313
System (SYSTEM) ........................ 313
Expiration date for member (EXPDTE) ..... 314
Maximum members (MAXMBRS) ............. 314
Access path size (ACCPTHSZ) .............. 314
Access path logical page size (PAGESIZE) .... 315
Access path type (ACCPTH) ............... 315
Access path maintenance (MAINT) ........... 315
Access path recovery (RECOVER) .......... 316
Force keyed access path (FRCACCPTH) ..... 316
Member size (SIZE) .................... 316
Allocate storage (ALLOCATE) ............. 317
Preferred storage unit (UNIT) ............. 318
Records to force a write (FRCRCATIO) ..... 318
Maximum file wait time (WAITFILE) ........ 318
Maximum record wait time (WAITRCD) ....... 319
Share open data path (SHARE) ............. 319
Max % deleted records allowed (DLTPCT). 319
Coded character set ID (CSSID) .......... 320
Allow update operation (ALWUPD) ......... 320
Allow delete operation (ALWDLT) ........... 320
Authority (AUT) ........................ 320
Examples ............................... 321
Error messages ........................ 321

Create Service Configuration
(CRTSRVCFG) .......................... 323
Parameters .......................... 323
Role (ROLE) .......................... 324
Connection type (CNNTYPE) ............... 324
Country or region ID (CNTRYID) .......... 325
State or province code (STATE) .......... 325
Primary telephone number (TELNBRI) ....... 325
Alternate telephone number (TELNBRII) .... 326
Resource name (SRCNAME) ............... 326
Remote system (RMTSYS) ................. 327
Proxy server (PROXY) .................. 327
ISP profile name (ISPPRF) ............... 328
Connectivity for others (CNNPNT) .......... 329
Connection point proxy (CNNPNTPRX) ....... 329
Examples ............................... 330
Error messages ........................ 331

Create Service Program
(CRTSRVPGM) .......................... 333
Parameters .......................... 333
Service program (SRVPGM) ............... 334
Module (MODULE) ........................ 334
Export (EXPORT) ........................ 335
Source file (SRCFILE) .................. 335
Source member (SRCMBR) ................ 336
Text ‘description’ (TEXT) ............... 336
Bind service program (BNDSRVPGM) ....... 336
Binding directory (BNDDIR) ............... 337
Activation group (ACTGRP) ............... 337
Creation options (OPTION) ............... 338
Listing detail (DETAIL) ................. 339
Allow update (ALWUPD) .................. 339
Allow *SRVPGM library update (ALWLUPD) .... 339
User profile (USRPRF) .................. 339
Replace program (REPLACE) .............. 340
Authority (AUT) ........................ 340
Target release (TGTRLS) ................ 340
Allow reinitialization (ALWREINZ) ....... 341
Storage model (STGMDL) ................. 341
Interprocedural analysis (IPA) ............ 342
IPA control file (IPACTLFILE) .......... 342
Examples ............................... 342
Error messages ........................ 342

Create Tape Category (CRTTAPCGY) 343
Parameters .......................... 344
Category (CGY) ........................ 344
Examples ............................... 344
Error messages ........................ 344

Create Tape File (CRTTAPF) .............. 347
Parameters .......................... 347
File (FILE) .......................... 348
Device (DEV) .......................... 348
Volume identifier (VOL) ................. 349
Tape reels specifications (REELS) ......... 350
Sequence number (SEQNBR) ............... 351
Tape label (LABEL) .................... 352
File type (FILETYPE) .................. 352
User specified DBCS data (IGCDTA) ....... 352
Text ‘description’ (TEXT) ............... 352
Record length (RCDLEN) ................. 353
Block length (BLKLEN) ................. 353
Buffer offset (BUFOSET) ............... 354
Record block format (RCDBLKfmt) ......... 355
Extend file (EXTEND) .................. 356
Tape density (DENSITY) ................. 356
Data compaction (COMPACT) ............. 361
Code (CODE) .......................... 361
Creation date (CRDDATE) ............... 361
File expiration date (EXPDTE) .......... 361
End of tape option (ENDOPT) ............ 362
User label program (USRLBLPGM) ......... 362
Maximum file wait time (WAITFILE) ....... 362
Share open data path (SHARE) .......... 363
Authority (AUT) ........................ 363
Replace file (REPLACE) ................. 364
Examples ............................... 364
Error messages ........................ 364

Create Table (CRTTBL) .................. 365
Parameters .......................... 365
Table (TBL) .......................... 366
Source file (SRCFILE) .................. 366
Source member (SRCMBR) ................. 367
Table type (TBLTYPE) .................. 367
Create Time Zone Description (CRTTIMZON) .................................. 371
Parameters ......................................................... 371
Time zone description (TIMZON) .............................. 371
Offset (OFFSET) .................................................... 372
Standard Time (STDNAME) ....................................... 372
Daylight Saving Time (DST) (DSTNAME) .................... 373
Standard Time message (STDMSG) ............................ 373
Daylight Saving Time message (DSTMSG) ................... 374
Message file (MSGF) .............................................. 374
Daylight Saving Time start (DSTSTR) ....................... 374
Daylight Saving Time end (DSTEND) ......................... 375
Text ‘description’ (TEXT) ....................................... 377
Authority (AUT) ................................................. 377
Examples .......................................................... 377
Error messages ................................................... 378

Create User-Defined FS (CRTUDFS) .......................... 379
Parameters ......................................................... 379
User-defined file system (UDFS) ............................... 379
Public authority for data (DTAAUT) ......................... 380
Public authority for object (OBJAUT) ....................... 380
Auditing value for objects (CRTOBJAUD) .................... 381
Scanning option for objects (CRTOBJSCAN) ............... 382
Restricted rename and unlink (RSTDRNMUNL) .............. 383
Case sensitivity (CASE) .......................................... 383
Default file format (DFTFILEMT) .............................. 383
Text ‘description’ (TEXT) ....................................... 383
Examples .......................................................... 384
Error messages ................................................... 384

Create User Profile (CRTUSRPRF) .............................. 385
Parameters ......................................................... 385
User profile (USRPRF) ........................................... 385
User password (PASSWORD) ................................... 385
Set password to expired (PWDEXP) ......................... 386
Status (STATUS) .................................................. 387
User class (USRCLS) ............................................ 388
Assistance level (ASTLVL) ...................................... 389
Current library (CURLIB) ...................................... 389
Initial program to call (INLPGM) ............................. 390
Initial menu (INLMNU) ........................................... 390
Limit capabilities (LMTCPB) .................................... 390
Text ‘description’ (TEXT) ....................................... 390
Special authority (SPCAUT) .................................... 390
Special environment (SPCENV) ............................... 390
Display sign-on information (DSPSGNINF) .................. 390
Password expiration interval (PWDEXPITV) ................. 390
Local password management (LCLPWSMGT) ................ 390
Limit device sessions (LMTDEVSSN) ......................... 390
Keyboard buffering (KBDBUF) ................................ 390
Maximum allowed storage (MAXSTG) ....................... 395
Highest schedule priority (PTYLMT) ......................... 395
Job description (JOBID) ........................................ 395
Group profile (GPGRP) ......................................... 396
Owner (OWNER) .................................................. 396
Group authority (GRPAUT) ...................................... 397
Group authority type (GRPAUTTYPE) ......................... 397
Supplemental groups (SUPGRP) ............................... 398
Accounting code (ACCGDE) .................................... 398
Document password (DOCPWD) .............................. 399
Message queue (MSGQ) .......................................... 399
Delivery (DLVRY) ................................................ 400
Severity code filter (SEV) ...................................... 400
Print device (PRTDEV) .......................................... 400
Output queue (OUTQ) .......................................... 400
Attention program (ATNPGM) ............................... 401
Sort sequence (SORTSEQ) ..................................... 401
Language ID (LANGID) ......................................... 402
Country or region ID (CNTRYID) .............................. 402
Coded character set ID (CCSID) .............................. 402
Character identifier control (CHRIDCTL) .................... 403
Locale job attributes (SETJOBAT) ......................... 403
Locale (LOCALE) ................................................ 403
User options (USROPT) ........................................ 404
User ID number (UID) .......................................... 405
Group ID number (GID) ........................................ 405
Home directory (HOMEHDIR) .................................. 405
EIM association (EIMASSOC) ................................. 406
Authority (AUT) .................................................. 407
Examples .......................................................... 407
Error messages ................................................... 408

Create Validation List (CRTVLDL) ............................ 411
Parameters ......................................................... 411
Validation list (VLDL) .......................................... 411
Text ‘description’ (TEXT) ....................................... 411
Authority (AUT) .................................................. 412
Examples .......................................................... 412
Error messages ................................................... 412

Create WSCST (CRTWSCST) .................................... 415
Parameters ......................................................... 415
Workstation customizing object (WSCST) ................... 415
Source member (SRCMBR) ...................................... 415
Text ‘description’ (TEXT) ....................................... 415
Source file (SRCFILE) .......................................... 415
Authority (AUT) .................................................. 415
Replace object (REPLACE) ..................................... 415
Examples .......................................................... 415
Error messages ................................................... 415

Convert CL Source (CVTCLSRC) .............................. 419
Parameters ......................................................... 419
From file (FROMFILE) .......................................... 420
To file (TOFILE) .................................................. 420
Member (FROMMBR) ............................................. 420
Examples .......................................................... 421
Error messages ................................................... 421

Convert Date (CVTDAT) ......................................... 423
Parameters ................................. 423
Date to be converted (DATE) ................................. 423
CL var for converted date (TOVAR) ................................. 424
From date format (FROMFMT) ................................. 424
To date format (TOFMT) ................................. 425
To date separator (TOSEP) ................................. 426
Examples ........................................ 426
Error messages ................................. 427

Convert Directory (CVTDIR) ................................. 429
Parameters ................................. 429
Option (OPTION) ................................. 429
Run priority (RUNPTY) ................................. 430
File system (FILESYS) ................................. 430
Format (FORMAT) ................................. 430
Detail (DETAIL) ................................. 430
Auxiliary storage pool ID (ASP) ................................. 430
Examples ........................................ 430
Error messages ................................. 431

Convert DLS Name (CVTDLSNAM) ................................. 433
Parameters ................................. 433
Object (OBJ) ................................. 434
Directory subtree (SUBTREE) ................................. 434
Action (ACTION) ................................. 434
Preview results (PREVIEW) ................................. 435
From CCSID (FROMCCSID) ................................. 435
To CCSID (TOCCSID) ................................. 435
Examples ........................................ 436
Error messages ................................. 436

Convert Education (CVTEDU) ................................. 439
Parameters ................................. 439
Course ID (COURSE) ................................. 439
Language ID (LNG) ................................. 439
Examples ........................................ 440
Error messages ................................. 440

Convert IP Address (CVTIPSIFIC) ................................. 441
Parameters ................................. 441
Internet address (INTNETADR) ................................. 441
Output (OUTPUT) ................................. 441
Examples ........................................ 442
Error messages ................................. 442

Convert Network ID / Location (CVTIPSLOC) ................................. 443
Parameters ................................. 443
Network identifier (NETID) ................................. 443
Location name (LOC) ................................. 443
Output (OUTPUT) ................................. 444
Examples ........................................ 444
Error messages ................................. 444

Convert Optical Backup (CVTOPTBKU) ................................. 445
Parameters ................................. 445
Backup volume identifier (BKUVOL) ................................. 445
Primary volume identifier (PRIVOL) ................................. 445
Examples ........................................ 446
Error messages ................................. 446

Convert Performance Data (CVTPFRDTA) ................................. 449
Parameters ................................. 449
From library (FROMLIB) ................................. 449
To library (TOLIB) ................................. 449
Job description (JOBID) ................................. 450
Examples ........................................ 450
Error messages ................................. 450

Convert Pfr Thread Data (CVTPFRTHD) ................................. 453
Parameters ................................. 453
Member (MBR) ................................. 453
Library (LIB) ................................. 453
Replace (REPLACE) ................................. 454
Examples ........................................ 454
Error messages ................................. 454

Convert RPC Source (CVTRPCSRC) ................................. 455
Parameters ................................. 455
From file (FROMFILE) ................................. 455
Option (OPTION) ................................. 455
Protocol (PROTOCOL) ................................. 456
To file (TOFILE) ................................. 456
Examples ........................................ 457
Error messages ................................. 457

Convert TCP/IP CL Source (CVTTPCPL) ................................. 459
Parameters ................................. 460
From file (FROMFILE) ................................. 460
To file (TOFILE) ................................. 460
From member (FROMMBR) ................................. 461
Examples ........................................ 461
Error messages ................................. 461

Convert To Folder (CVTTOFLR) ................................. 463
Error messages for CVTTOFLR ................................. 463
Parameters ................................. 463
From virtual disk (FROMVDSK) ................................. 463
To folder (TOFLR) ................................. 464
Replace documents (REPLACE) ................................. 464
Examples ........................................ 464
Error messages ................................. 464

Convert User Certificate (CVTUSRCERT) ................................. 467
Parameters ................................. 467
User profile (USRPRF) ................................. 467
Option (OPTION) ................................. 467
Examples ........................................ 468
Error messages ................................. 468

Data (DATA) ................................. 469
Parameters ................................. 469

Contents xi
<table>
<thead>
<tr>
<th>Command</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Declare CL Variable (DCL)</td>
<td>473</td>
</tr>
<tr>
<td>Declare File (DCLF)</td>
<td>479</td>
</tr>
<tr>
<td>Declare Processing Options (DCLPRCOPT)</td>
<td>485</td>
</tr>
<tr>
<td>Decompress Object (DCPOBJ)</td>
<td>487</td>
</tr>
<tr>
<td>Remove Link (DEL)</td>
<td>493</td>
</tr>
<tr>
<td>Dependent Definition (DEP)</td>
<td>497</td>
</tr>
<tr>
<td>Create Object (DLCOBJ)</td>
<td>503</td>
</tr>
<tr>
<td>Delete Cluster Admin Domain (DLTADMMDMN)</td>
<td>509</td>
</tr>
<tr>
<td>Delete Alert (DLTALR)</td>
<td>511</td>
</tr>
<tr>
<td>Delete Alert Table (DLTALRBTBL)</td>
<td>515</td>
</tr>
<tr>
<td>Delete APAR Data (DLTAPARDTA)</td>
<td>519</td>
</tr>
<tr>
<td>Delete Authority Holder (DLTAUTHLR)</td>
<td>521</td>
</tr>
<tr>
<td>Delete Authorization List (DLTAUTHL)</td>
<td>523</td>
</tr>
<tr>
<td>Delete Binding Directory (DLTBNDDIR)</td>
<td>525</td>
</tr>
</tbody>
</table>
Binding directory (BNDDIR) .......................... 525
Examples ........................................... 526
Error messages ..................................... 526

Delete Configuration List (DLTCFGL) 529
Parameters ........................................... 529
Configuration list (CFGL) .......................... 529
Examples ........................................... 529
Error messages ..................................... 529

Delete C Locale Description (DLTCLD) 531
Error messages for DLTCLD ......................... 531
Parameters ........................................... 531
Locale name (CLD) ................................... 531
Examples ........................................... 532
Error messages ..................................... 532

Delete Class (DLTCLS) .............................. 533
Parameters ........................................... 533
Class (CLS) ......................................... 533
Examples ........................................... 534
Error messages ..................................... 534

Delete Cluster (DLTCLU) ......................... 537
Parameters ........................................... 537
Cluster (CLUSTER) ................................... 537
Examples ........................................... 538
Error messages ..................................... 538

Delete Command (DLTCMD) ....................... 539
Parameters ........................................... 539
Command (CMD) ...................................... 539
Examples ........................................... 540
Error messages ..................................... 540

Delete Communications Trace
(DLTCMNTRC) ....................................... 543
Parameters ........................................... 543
Configuration object (CFGOBJ) ................. 543
Type (CFGTYPE) ..................................... 543
Examples ........................................... 544
Error messages ..................................... 544

Delete Connection List (DLTCNNL) ............ 545
Parameters ........................................... 545
Connection list (CNNL) ............................ 545
Examples ........................................... 545
Error messages ..................................... 546

Delete Class-of-Service Desc
(DLTCOSD) ......................................... 547
Parameters ........................................... 547
Class-of-service description (COSD) .......... 547
Examples ........................................... 547
Error messages ..................................... 547

Delete Cluster Resource Group
(DLTCRG) ........................................... 549
Parameters ........................................... 549
Cluster resource group (CRG) .................... 549
Examples ........................................... 550
Error messages ..................................... 550

Delete CRG Cluster (DLTCRGCLU) ............. 553
Parameters ........................................... 553
Cluster (CLUSTER) ................................... 554
Cluster resource group (CRG) .................... 554
Examples ........................................... 554
Error messages ..................................... 554

Delete CRQ Description (DLTCRQD) ............ 555
Parameters ........................................... 555
Change request description (CRQD) .......... 555
Examples ........................................... 556
Error messages ..................................... 556

Delete Comm Side Information
(DLTCSSI) ........................................... 557
Parameters ........................................... 557
Side information (CSI) ............................ 557
Examples ........................................... 558
Error messages ..................................... 558

Delete Controller Description
(DLTCCTL) .......................................... 559
Parameters ........................................... 559
Controller description (CTLD) .................. 559
Examples ........................................... 559
Error messages ..................................... 559

Delete Device Description (DLTDEVD) ...... 561
Parameters ........................................... 561
Device description (DEVD) ....................... 561
Examples ........................................... 561
Error messages ..................................... 561

Delete Device Media Library
(DLTDEVMLB) ....................................... 563
Parameters ........................................... 563
Device description (DEVD) ....................... 563
Examples ........................................... 563
Error messages ..................................... 563

Delete DFU Program (DLTDFUPGM) ............. 565
Error messages for DLTDFUPGM ................. 565
Parameters ........................................... 565
DFU program (DFUPGM) ............................ 565
Examples ........................................... 565
Error messages ..................................... 565

Delete Document Library Object
(DLTDLLO) .......................................... 567
Parameters ........................................... 567
Document library object (DLO) .................. 568
In folder (FLR) ...................................... 569
Creation period (CRTDATE) ....................... 569
Create Message File (CRTMSGF)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Create Message File (CRTMSGF) command creates a user-defined message file for storing message descriptions. The message file should be stored in a library for which all users who are to use the predefined messages have authority. The system is shipped with the IBM-supplied message files, stored in the system library, QSYS: the CPF message file, QCPFMSG (for the OS/400 system and machine interface messages); and the licensed program message files, such as QRPMSG (for RPG messages).

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSGF</td>
<td>Message file</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Message file</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>TEXT</td>
<td>Text ‘description’</td>
<td>Character value, *BLANK</td>
<td>Optional</td>
</tr>
<tr>
<td>SIZE</td>
<td>File size</td>
<td>Element list</td>
<td>Optional, Positional 2</td>
</tr>
<tr>
<td></td>
<td>Element 1: Initial storage size</td>
<td>Integer, 10</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Increment storage size</td>
<td>Integer, 2</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 3: Maximum increments</td>
<td>Integer, *NOMAX</td>
<td></td>
</tr>
</tbody>
</table>

Message file (MSGF)

Specifies the message file to be created.

This is a required parameter.

Qualifier 1: Message file

_message-file-name_

Specify the name of the message file being created.

Qualifier 2: Library

*CURLIB

The current library for the job is used to create the message file. If no current library entry exists in the library list, the QGPL library is used.
library-name
   Specify the library where the message file is to be created.

Text ’description’ (TEXT)
   Specifies the text that briefly describes the object.
   *BLANK
      No text is specified.
   ’description’
      Specify no more than 50 characters, enclosed in apostrophes.

File size (SIZE)
   Specifies the initial storage size of the message file, the amount of each increase in its storage, and the number of times the size can be increased. The storage size is expressed in kilobytes (KB). (1KB equals 1024 bytes of storage.)

Element 1: Initial storage size
   10
   The message file has 10 KB of storage assigned to it.
   initial-Kilobytes
      Specify the initial size of the file (must be greater than 0).

Element 2: Increment storage size
   2
   2 KB of storage is added to the message file each time its size is increased.
   increment-value
      Specify the number of kilobytes added each time the message file’s size is increased.

Element 3: Maximum increments
   *NOMAX
      The amount added to the message file is not limited by the user. The maximum size is determined by the system.
   number-of-increments
      Specify the maximum number of times that a message file’s size can be increased. Specify a 0 to prevent any additions to the initial size of the file.

Authority (AUT)
   Specifies the authority you are giving to users who do not have specific authority for the object, who are not on an authorization list, and whose group profile or supplemental group profiles do not have specific authority for the object.
   *LIBCRTAUT
      The system determines the authority for the object by using the value specified for the Create authority (CRTAUT) parameter on the Create Library command (CRTLIB) for the library.
containing the object to be created. If the value specified for the CRTAUT parameter is changed, the new value will not affect any existing objects.

*CHANGE
The user can perform all operations on the object except those limited to the owner or controlled by object existence (*OBJEXIST) and object management (*OBJMGT) authorities. The user can change and perform basic functions on the object. *CHANGE authority provides object operational (*OBJOPR) authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove users.

*ALL
The user can perform all operations except those limited to the owner or controlled by authorization list management (*AUTLMGT) authority. The user can control the object’s existence, specify the security for the object, change the object, and perform basic functions on the object. The user also can change ownership of the object.

*USE
The user can perform basic operations on the object, such as running a program or reading a file. The user cannot change the object. Use (*USE) authority provides object operational (*OBJOPR), read (*READ), and execute (*EXECUTE) authorities.

*EXCLUDE
The user cannot access the object.

name  Specify the name of an authorization list to be used for authority to the object. Users included in the authorization list are granted authority to the object as specified in the list. The authorization list must exist when the object is created.

Coded character set ID (CCSID)

Specifies the coded character set identifier (CCSID) that is to be associated with the message file. The CCSID associated with the message file always overrides the CCSID associated with the message description. To use the CCSID associated with the message description, change the CCSID associated with the message file to *MSGD. For more information on message handler and its use of CCSIDs, see the Globalization topic in the iSeries Information Center at http://www.ibm.com/eserver/iseries/infocenter.

*HEX
The CCSID that is associated with the message file is set to 65535. The 65535 CCSID means that no conversions are to occur when adding or changing message descriptions in the message file and no conversions are to occur when retrieving message descriptions from the file. The CCSID associated with the message description is saved in the event the message file is ever changed to *MSGD.

*MSGD
The CCSID that is associated with the message file is set to 65534. The 65534 CCSID means to use the CCSID associated with the message description when retrieving message text from the file. When adding or changing message descriptions in the message file, no conversions are to occur. The message description is tagged with the CCSID specified on the ADDMSGD or CHGMSGD commands.

*JOB
The CCSID that is associated with the message file is the CCSID of the job calling this command.

coded-character-set-identifier
Specify the CCSID that the message file is to be created with. Any message descriptions added to this message file are converted from the CCSID specified to the CCSID of the message file. Valid values range from 1 through 65535. See the Globalization information in the iSeries Information Center at http://www.ibm.com/eserver/iseries/infocenter for a list of valid CCSID values. Only CCSID values that a job can be changed to are accepted.
Examples

This command creates a message file named INVMSG in which predefined inventory application messages are stored. The file is stored in the library INVLIB, for which all users of the file must have *USE authority. Because the AUT parameter is defaulted, all users have *CHANGE authority for the file, meaning they can retrieve messages from the file.

Error messages

*ESCAPE Messages

CPF2108
Object &1 type * &3 not added to library &2.

CPF2112
Object &1 in &2 type * &3 already exists.

CPF2113
Cannot allocate library &1.

CPF2151
Operation failed for &2 in &1 type * &3.

CPF2182
Not authorized to library &1.

CPF2283
Authorization list &1 does not exist.

CPF2402
Library &1 not found

CPF247E
CCSID &1 is not valid.

CPF2497
Size for &1 in &2 exceeds machine limit.

CPF9838
User profile storage limit exceeded.
Create Menu from Msg Files (CRTMSGFMNU)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Create Message File Menu (CRTMSGFMNU) command creates a menu (display file) from the specified message files. You can specify that this menu is created either in a fixed-format, with options 1 through 24 arranged in two columns, or in free-format.

If you want to create a menu from the source member, use the Create System/36 Menu (CRTS36MNU) command.

Restriction: Option 5 of the operating system must be installed to run this command. This command can be run either natively or in the System/36 environment.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMDTXTMSGF</td>
<td>Menu## command message file</td>
<td>Qualified object name</td>
<td>Optional, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Menu## command</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>message file</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>OPTTXTMSGF</td>
<td>Option text message file</td>
<td>Single values: *NONE</td>
<td>Optional, Positional 2</td>
</tr>
<tr>
<td></td>
<td>Other values: Qualified object</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>name</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Option text</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>message file</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *CMDLIB, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>REPLACE</td>
<td>Replace menu</td>
<td>*NO, *YES</td>
<td>Optional</td>
</tr>
<tr>
<td>FREEFORM</td>
<td>Free form menu</td>
<td>*NO, *YES</td>
<td>Optional</td>
</tr>
<tr>
<td>DDSLIST</td>
<td>DDS listing</td>
<td>*PARTIAL, *FULL</td>
<td>Optional</td>
</tr>
<tr>
<td>MAXDEV</td>
<td>Maximum devices</td>
<td>1-256, 5</td>
<td>Optional</td>
</tr>
<tr>
<td>AUT</td>
<td>Authority</td>
<td>Name, *USE, *ALL, *CHANGE, *EXCLUDE,</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>*LIBCRTAUT</td>
<td></td>
<td></td>
</tr>
<tr>
<td>TOFILE</td>
<td>To DDS source file</td>
<td>Qualified object name</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: To DDS source file</td>
<td>Name, QDDSSRC</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *CMDLIB, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>TOMBR</td>
<td>To DDS source member</td>
<td>Name, *NONE</td>
<td>Optional</td>
</tr>
<tr>
<td>IGCDTA</td>
<td>User specified DBCS data</td>
<td>*NO, *YES</td>
<td>Optional</td>
</tr>
<tr>
<td>TGTRLS</td>
<td>Target Release</td>
<td>Character value, *CURRENT, *PRV</td>
<td>Optional</td>
</tr>
</tbody>
</table>
Menu## command message file (CMDTXTMSGF)

Specifies the name and library of the command text message file that contains the text for the command that runs when the corresponding option is selected. This parameter must identify an existing message file and not a screen file generator (SFGR) or a System/36 message source member. The message IDs used must begin with USR. Trailing ## symbols are required on the CMDTXTMSGF name. The menu (display file) name is the message-file-name without the ## symbols. The run-time menu processor appends the trailing ## symbols to the menu name to determine the message file name. This is the name of the message file that contains messages whose text is the command that is run for any option selected.

This is a required parameter.

message-file-name-##
   Specify the message file name used to create the menu. The trailing ## symbols are required. The menu (display file) name is the same as the message file name without the ## symbols.

The possible library values are:

*CURLIB
   The current library for the job is used to locate the file. If no library is specified as the current library for the job, the QGPL library is used.

library-name
   Specify the name of the library where the command text message file is located. The CMDTXTMSGF library is also where the menu display file is created. This is a requirement of the run-time MENU processor.

Option text message file (OPTTXTMSGF)

Specifies the name and library of the option text message file that contains the text that is displayed on the menu to describe the options that can be selected. The message IDs used must begin with USR, unless *YES is specified on the User specified DBCS data (IGCDTA) parameter, which allows the message ID to begin with USZ.

*NONE
   No option text message file is used. The Menu## command message file (CMDTXTXTMSGF) parameter is used to specify the option text.

message-file-name
   Specify the name of the option text message file used for the descriptions of the options on the menu you are creating.

The possible library values are:

*CMDLIB
   The library specified on the Menu## command message file (CMDTXTXTMSGF) parameter is used to locate the option text message file.

*CURLIB
   The current library for the job is used to locate the option text message file. If no library is specified as the current library for the job, the QGPL library is used.

library-name
   Specify the name of the library where the option text message file is located.
Replace menu (REPLACE)

Specifies whether the original display file is replaced by the new file.

*NO    The existing display file does not change.

*YES   The existing display file is replaced by the one being created. Other types of files are not replaced.

Note: The menu is not created if it has the same name and library as an existing program or message file.

If terminating errors are encountered, the existing display file is not replaced. If the display file already exists, the Authority (AUT) parameter is ignored and the authorities for the old display file are copied to the new menu display file that replaces it.

Free form menu (FREEFORM)

Specifies whether the menu is created in free-format or in a fixed-format.

*NO    Free-format is not used. A fixed-format menu with two-columns is created. Option text message numbers correspond to the option numbers.

*YES   A free-format menu is created. If *YES is specified, a value must be specified on the Option text message file (OPTTXTMSGF) parameter. Option text message numbers correspond to the row numbers on the display.

DDS listing (DDSLIST)

Specifies whether a partial or full DDS compile listing is provided.

*PARTIAL A partial listing is provided.

*FULL   A full DDS listing and cross-reference are provided.

Maximum devices (MAXDEV)

Specifies the maximum number of devices that can use the menu at one time.

5      The maximum number of devices is five.

number-of-devices Specify the maximum number of devices that can use the menu at one time. Valid values range from 1 through 256.

Authority (AUT)

Specifies the authority you are giving the users who do not have specific authority to the object, who are not on the authorization list, and whose user group has no specific authority to the object.
If OPTION(*CREATE) and REPLACE(*YES) are specified, and the display file already exists, the AUT parameter is ignored and the authorities for the old display file are copied to the new display file that replaces it.

**LIBCRTAUT**

- The authority for the object is the same as the value specified on the Create authority (CRTAUT) parameter of the library in which the object is being created. If the value specified on the CRTAUT parameter is changed, the new value will not affect any existing objects.

**USE**

- The user can perform basic operations on the object, such as running a program or reading a file. The user cannot change the object. Use (*USE) authority provides object operational (*OBJOPR), read (*READ), and execute (*EXECUTE) authorities.

**ALL**

- The user can perform all operations except those limited to the owner or controlled by authorization list management (*AUTLMGT) authority. The user can control the object’s existence, specify the security for the object, change the object, and perform basic functions on the object. The user also can change ownership of the object.

**CHANGE**

- The user can perform all operations on the object except those limited to the owner or controlled by object existence (*OBJEXIST) and object management (*OBJMGT) authorities. The user can change and perform basic functions on the object. *CHANGE authority provides object operational (*OBJOPR) authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove users.

**EXCLUDE**

- The user cannot access the object.

**authorization-list-name**

- Specify the name of an authorization list to be used for authority to the object. Users included in the authorization list are granted authority to the object as specified in the list. The authorization list must exist when the object is created.

---

**To DDS source file (TOFILE)**

Specifies the name and library of the source file in which to store the DDS source that is used to create the menu display. The source file need not already exist. If the user is authorized to the CRTSRCBF (Create Source Physical File) command, and the file does not exist, a new source file is created. This parameter is ignored if TOMBR(*NONE) is specified.

**QDDSSRC**

- The source file, QDDSSRC, is used.

**file-name**

- Specify the name of the source file in which to store the DDS source.

The possible library values are:

**CMDLIB**

- The library specified on the Menu## command message file (CMDDXTMSGF) parameter is used to locate the source file.
*CURLIB

The current library for the job is used to locate the file. If no current library entry exists in the library list, the QGPL library is used.

library-name

Specify the name of the library where the source file is located.

To DDS source member (TOMBR)

Specifies the source file member name in which to store the DDS source. If the member does not exist, it is created. When the member name is the same as that of the display file name, and the to-file is QS36DDSSRC in the same library as the display file being created, the DDS is saved in this member only if the compile operation of the display file is successful. To guarantee that the DDS is saved, specify the name of some other source file, library, or member.

*NONE

The DDS source is not stored in the source file specified on the To DDS source file (TOFILE) parameter.

member-name

Specify the name of the source file member in which to store the DDS source. If the member does not exist, it is added. If it exists, it is replaced.

User specified DBCS data (IGCDTA)

Specifies whether the display file contains double-byte character data.

*NO

The display file does not contain double-byte character data. Option text message IDs must begin with USR.

*YES

The display file or the message files contain double-byte character data. Option text message IDs can begin with USZ.

Target Release (TGTRLS)

Specifies the release level of the operating system on which you intend to use the object being saved.

When specifying the target-release value, the format VxRxMx is used to specify the release, where Vx is the version, Rx is the release, and Mx is the modification level. For example, V5R3M0 is version 5, release 3, modification 0.

Valid values depend on the current version, release, and modification level of the operating system, and they change with each new release. You can press F4 while prompting this command parameter to see a list of valid target release values.

*CURRENT

The object is to be restored to, and used on, the release of the operating system currently running on your system. The object can also be restored to a system with any subsequent release of the operating system installed.

Create Menu from Msg Files (CRTMSGFMNU)
*PRV  The object is to be restored to the previous release with modification level 0 of the operating system. The object can also be restored to a system with any subsequent release of the operating system installed.

**character-value**

Specify the release in the format VxRxMx. The object can be restored to a system with the specified release or with any subsequent release of the operating system installed.

---

### Examples

**CRTMSGFMNU CMDTXTMSGF(MYMENU##) REPLACE(*YES) FREEFORM(*NO)**

This command creates a menu by using a message file named MYMENU. The message file is located in the current library for the job and it contains the commands run for each menu option. Because no OPTTXTMSGF parameter is specified, the text of the command appears on the screen in place of the option text. REPLACE(*YES) specifies that an existing display file is replaced. The created display file is in a fixed-format, with options 1 through 24 arranged in two 12-element columns.

---

### Error messages

***ESCAPE Messages**

SSP4464

Member &3 in file &1 in use, cannot be shared.

SSP5004

&1—This load member exists, but is not a $SFGR member.

SSP5005

&1 display file already exists.

SSP5011

&1 not allowed for display file name.

SSP5017

TOFILE library &1 not found.

SSP5019

Terminating errors in $SFGR input specifications.

SSP5027

TGTRLS(*PRV) allowed with changes only when existing display file created for previous release.

SSP5451

Existing file &1 is not a display file.

SSP5750

Command message file messages 1-24 contain only blank text.

SSP5751

Command text message file name must end with ##.

SSP5752

Command text message file library &1 not found.

SSP5753

Command text message file &1 not found.
SSP5754
Option text message file &1 not found.
SSP5755
Unable to create $BMENU work file.
SSP5756
Command message file name must be longer than 2 characters.
SSP5757
Command text message file has no MIC in 0001-0024 range
SSP5762
Option text message file name cannot be same as menu name.
SSP5770
Option text message file required for free format menu.
SSP5774
Command and option message files must not be the same.
SSP6124
Unexpected error occurred.
SSP7375
Error &1 received by &2 utility.
SSP8663
User not authorized to access &1.
SSP8679
Not authorized to access member &1.
Create Message Queue (CRTMSGQ)

Where allowed to run: All environments (*ALL)
Threadsafe: Yes

The Create Message Queue (CRTMSGQ) command creates a user-defined message queue and stores it in a specified library. The message queue should be put in a library for which all users who are to send messages to and receive messages from the queue have *USE authority. The messages sent can be either predefined messages or immediate messages. The message queue has the following attributes initialized when it is created: the DLVRY parameter is set to *HOLD, the first element of the PGM parameter is set to *DSPMSG and the second element of the PGM parameter is set to *ALWRPY, SEV is set to 00, and RESET is set to *NO. These initialized attributes cannot be specified on the CRTMSGQ command and the CHGMSGQ command must be used to change them after the queue is created.

Note: Message queue QSYSOPR is shipped with a message queue full action of *WRAP. If the value is changed to *SNDMSG and the queue needs to be recreated because it was damaged, the value is reset to the shipped value of *WRAP.

### Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSGQ</td>
<td>Message queue</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Message queue</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>TEXT</td>
<td>Text ‘description’</td>
<td>Character value, *BLANK</td>
<td>Optional</td>
</tr>
<tr>
<td>FORCE</td>
<td>Force to auxiliary storage</td>
<td>*NO, *YES</td>
<td>Optional</td>
</tr>
<tr>
<td>SIZE</td>
<td>Queue size</td>
<td>Element list</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Element 1: Initial storage</td>
<td>Integer, 3</td>
<td></td>
</tr>
<tr>
<td></td>
<td>size</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Increment storage</td>
<td>Integer, 1</td>
<td></td>
</tr>
<tr>
<td></td>
<td>size</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 3: Maximum increments</td>
<td>Integer, *NOMAX</td>
<td></td>
</tr>
<tr>
<td>ALWALR</td>
<td>Allow alerts</td>
<td>*NO, *YES</td>
<td>Optional</td>
</tr>
<tr>
<td>CCSID</td>
<td>Coded character set ID</td>
<td>1-65535, *MSG, *HEX, *JOB</td>
<td>Optional</td>
</tr>
<tr>
<td>MSGQFULL</td>
<td>Message queue full action</td>
<td>*SNDMSG, *WRAP</td>
<td>Optional</td>
</tr>
</tbody>
</table>
Message queue (MSGQ)

Specifies the message queue to be created.

This is a required parameter.

Qualifier 1: Message queue

name Specify the name of the message queue being created.

Qualifier 2: Library

*CURLIB

The current library for the job is used to create the message queue. If no current library entry exists in the library list, the QGPL library is used.

name Specify the library where the message queue is to be created.

Text ’description’ (TEXT)

Specifies the text that briefly describes the object.

*BLANK

No text is specified.

’description’

Enter no more than 50 characters, enclosed in apostrophes.

Force to auxiliary storage (FORCE)

Specifies whether changes made to the message queue description or messages added to or removed from the queue are immediately forced into auxiliary storage; this ensures that changes to the queue, or messages sent or received, are not lost if a system failure occurs.

*NO Changes made to the message queue, including its messages, are not immediately forced to auxiliary storage.

*YES All changes to the message queue description and to the messages in the queue are immediately forced to auxiliary storage.

Queue size (SIZE)

Specifies the initial storage size of the message queue, the size of each addition to its storage, and the number of times the size can be increased. The storage size is expressed in kilobytes (KB).

Element 1: Initial storage size

3 Initially, the message queue has 3 KB of storage assigned to it. (1 KB equals 1024 bytes of storage.)

initial-Kilobytes Specify the initial size of the queue (must be greater than 0).
Element 2: Increment storage size

One of the following is used to specify the amount of storage in kilobytes added to the message queue’s size each time the size is increased.

1 One KB of storage is added to the message queue each time its size is increased.

*increment-value

Specify the number of kilobytes added each time the message queue’s size is increased.

Element 3: Maximum increments

One of the following is used to specify the maximum number of times the message queue’s size can be increased.

*NOMAX

The number of times storage can be added to the message queue is not limited by the user. The maximum size is determined by the system.

*number-of-increments

Specify the maximum number of times storage can be added to the queue. Enter a 0 to prevent any additions to the initial size of the queue.

------

Authority (AUT)

Specifies the authority you are giving to users who do not have specific authority for the object, who are not on an authorization list, and whose group profile or supplemental group profiles do not have specific authority for the object.

*LIBCRTAUT

The system determines the authority for the object by using the value specified for the Create authority (CRTAUT) parameter on the Create Library command (CRTLIB) for the library containing the object to be created. If the value specified for the CRTAUT parameter is changed, the new value will not affect any existing objects.

*CHANGE

The user can perform all operations on the object except those limited to the owner or controlled by object existence (*OBJEXIST) and object management (*OBJMGT) authorities. The user can change and perform basic functions on the object. *CHANGE authority provides object operational (*OBJOPR) authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove users.

*ALL

The user can perform all operations except those limited to the owner or controlled by authorization list management (*AUTLMGT) authority. The user can control the object’s existence, specify the security for the object, change the object, and perform basic functions on the object. The user also can change ownership of the object.

*USE

The user can perform basic operations on the object, such as running a program or reading a file. The user cannot change the object. Use (*USE) authority provides object operational (*OBJOPR), read (*READ), and execute (*EXECUTE) authorities.

*EXCLUDE

The user cannot access the object.

name

Specify the name of an authorization list to be used for authority to the object. Users included in the authorization list are granted authority to the object as specified in the list. The authorization list must exist when the object is created.
Allow alerts (ALWALR)

Specifies whether the queue being created allows alerts to be generated from alert messages that are sent to it.

*NO  Does not allow alerts to be generated from this message queue.

*YES  Allows alerts to be generated from this message queue.

Coded character set ID (CCSID)

Specifies the coded character set identifier (CCSID) associated with this message queue. The CCSID applies only to immediate messages and message data that is defined as a character field that can be converted (*CCHAR).

*HEX  Messages sent to, received from, or displayed from this message queue are not converted. The message queue CCSID is 65535.

*MSG  Messages sent to this message queue are not converted. The CCSID specified by the sending job is saved in case a conversion is needed for a display or receive function. The message queue CCSID is 65534.

*JOB  The CCSID of the message queue will be the CCSID of the job running this command.

coded-character-set-identifier

Specify the CCSID associated with this message queue. Messages sent to this message queue are converted to this CCSID. Valid values range from 1 through 65535. See the Globalization information in the iSeries Information Center at http://www.ibm.com/eserver/iseries/infocenter for a list of valid CCSID values.

For more information about the message handler and its use of CCSIDs, see the Globalization topic in the iSeries Information Center at http://www.ibm.com/eserver/iseries/infocenter.

Message queue full action (MSGQFULL)

Specifies the action to take when the message queue is full.

*SNDMSG  When the message queue is full, CPF2460 (Message queue could not be extended.) is sent to the program or user that is sending a message to the full message queue.

*WRAP  When the message queue is full, the oldest informational and answered messages are removed from the message queue to allow space for new messages to be added. If the removing of the informational and answered messages does not provide enough space to add the requested message, then unanswered inquiry messages are removed until there is space to add the requested message. The default reply is sent before an unanswered inquiry message is removed. When the message queue is wrapped, CPI2420 or CPI2421 will be sent to the queue that was full to indicate it was wrapped. If there is no space on the queue to send these messages they are sent to the joblog of the user that was sending the message to the queue and they are sent to QHST if the full queue is QSYSOPR.

NOTE:
When a queue uses *WRAP and a job sends a message to the queue that causes a wrap, messages are removed for the following conditions in order to perform the wrap:

- the queue is in break or notify mode for a job
- a job is in a message wait state because it did a receive function on the queue with a wait time specified
- the queue is allocated by a job via the ALCOBJ command

Only the system wrap function can remove messages from queues in these conditions. Other jobs still are not allowed to remove messages from the queues during these conditions. With *SNDMSG, these conditions do not allow another job to remove messages from the queue.

Also when a queue specifies *WRAP and it is in break mode, the wrap function only removes messages that have been received by the break-handling program. For example, if the break-handling program did not receive all messages from the queue and it was becoming full, CPF2460 could be issued because messages could not be removed to perform the wrap.

Examples

```
CRTMSGQ  MSGQ(MYQ)  SIZE(3 3 +NOMAX)
   TEXT('Message queue for inventory transactions')
   AUT(*CHANGE)
```

This command creates the message queue MYQ and stores it in the current library (*CURLIB) by default. All users are authorized to send messages to the queue and to read its messages.

The message queue is created with an initial size of 3 kilobytes (KB) and increased in size in 3 KB increments. The restriction on its maximum size is the system limit for objects, which is about 16,000 KB.

Error messages

**ESCAPE Messages**

CPF2108
Object &1 type *&3 not added to library &2.

CPF2112
Object &1 in &2 type *&3 already exists.

CPF2113
Cannot allocate library &1.

CPF2151
Operation failed for &2 in &1 type *&3.

CPF2182
Not authorized to library &1.

CPF2283
Authorization list &1 does not exist.

CPF2402
Library &1 not found

CPF247E
CCSID &1 is not valid.
CPF2497
Size for &1 in &2 exceeds machine limit.

CPF9838
User profile storage limit exceeded.
Create Node Group (CRTNODGRP)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Create Node Group (CRTNODGRP) command creates a node group to be used for creating distributed database files.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>NODGRP</td>
<td>Node group</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Node group</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>RDB</td>
<td>Relational database</td>
<td>Values (up to 32 repetitions): Character value</td>
<td>Required, Positional 2</td>
</tr>
<tr>
<td>PTNFILE</td>
<td>Partitioning file</td>
<td>Qualified object name</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Partitioning file</td>
<td>Name, *NONE</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>PTNMBR</td>
<td>Partitioning member</td>
<td>Name, *FIRST</td>
<td>Optional</td>
</tr>
<tr>
<td>TEXT</td>
<td>Text ‘description’</td>
<td>Character value, *BLANK</td>
<td>Optional</td>
</tr>
</tbody>
</table>

Node group (NODGRP)

Specifies the node group to be created.

This is a required parameter.

Qualifier 1: Node group

name Specify the name of the node group to be created.

Qualifier 2: Library

*CURLIB

The node group is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

name Specify the name of the library for the node group.
Relational database (RDB)

Specifies the relational databases to be included in the node group.

This is a required parameter.

character-value

Specify the name of each relational database to be used. A maximum of 18 characters can be specified for the relational database name. These must have already been defined in the system’s relational database directory using the Add RDB Directory Entry (ADDRDBDIRE) command. At least two relational database names must be specified. One of the entries must correspond to the local system. All other entries must correspond to remote iSeries systems. Up to 32 relational database names may be specified.

When the node group is created, a node number is assigned for each relational database specified. Node numbers are assigned consecutively, starting with 1. The first relational database is assigned node number 1, the second database is assigned node number 2, and so on. Once the node group has been created, you can use the DSPNODGRP (Display Node Group) command to see the correspondence between node numbers and relational database names.

Partitioning file (PTNFILE)

Specifies the name of the file to be used to determine the partitioning attributes for the node group. The node group contains a table with 1024 partitions. Each partition contains a node number. The partitioning file allows you to set the node number for each of the 1024 partitions. For a complete description of the format of a partitioning file, refer to the DB2 Multisystem for OS/400 book.

Qualifier 1: Partitioning file

*NONE

A partitioning file will not be used to set the partitioning attributes for the node group object.

Each valid node number will be assigned to equal number of partitions. For example, if two relational databases are specified there will be two valid node numbers (1 and 2), and the partitions will be divided equally so that 512 partitions have a node number of 1 and the other 512 partitions have a node number of 2.

name Specify the name of the partitioning file to be used.

Qualifier 2: Library

*LIBL All libraries in the library list for the current thread are searched until the first match is found.

*CURLIB

The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

name Specify the name of the library to be searched.

Partitioning member (PTNMBR)

Specifies the member in the partitioning file to be used to determine the partitioning attributes for the node group.

*FIRST

The first member in the partitioning file is used.
name Specify the name of the member to be used.

Note: This parameter is not valid when *NONE is specified for the Partitioning file (PTNFILE) parameter.

Text ’description’ (TEXT)

Specifies the text that briefly describes the node group.

*BLANK Text is not specified.

character-value Specify no more than 50 characters of text, enclosed in apostrophes.

Authority (AUT)

Specifies the authority you are giving to users who do not have specific authority for the object, who are not on an authorization list, and whose group profile or supplemental group profiles do not have specific authority for the object.

*LIBCRTAUT The system determines the authority for the object by using the value specified for the Create authority (CRTAUT) parameter on the Create Library command (CRTLIB) for the library containing the object to be created. If the value specified for the CRTAUT parameter is changed, the new value will not affect any existing objects.

*CHANGE The user can perform all operations on the object except those limited to the owner or controlled by object existence (*OBJEXIST) and object management (*OBJMGT) authorities. The user can change and perform basic functions on the object. *CHANGE authority provides object operational (*OBJOPR) authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove users.

*ALL The user can perform all operations except those limited to the owner or controlled by authorization list management (*AUTLMGT) authority. The user can control the object’s existence, specify the security for the object, change the object, and perform basic functions on the object. The user also can change ownership of the object.

*USE The user can perform basic operations on the object, such as running a program or reading a file. The user cannot change the object. Use (*USE) authority provides object operational (*OBJOPR), read (*READ), and execute (*EXECUTE) authorities.

*EXCLUDE The user cannot access the object.

name Specify the name of an authorization list to be used for authority to the object. Users included in the authorization list are granted authority to the object as specified in the list. The authorization list must exist when the object is created.

Examples

Example 1: Creating a Group with Default Partitioning
This command creates a node group containing four nodes. The partitioning attributes default to assigning one-fourth of the partitions to each node number. This node group can be used on the NODGRP parameter of the Create Physical File (CRTPF) CL command to create a distributed file. Distributed files created specifying this node group will have their data spread across the four node systems. If the records in the distributed file contain a uniform distribution of values for those fields which comprise the partition key, the records will be spread evenly between the node systems.

**Example 2: Creating a Group with Specific Partitioning**

```
CRTNODGRP NODGRP(LIB1/GROUP2)
   RDB(SYSTEMA SYSTEMB SYSTEMC SYSTEMD)
   PTNFILE(LIB1/PTN1)
   TEXT('Partition most of the data to SYSTEMA')
```

This command creates a node group containing three nodes. The partitioning attributes are taken from the file called PTN1. This file can be set up to force a higher percentage of the records (or rows) to be located on a particular system.

---

**Error messages**

*ESCAPE Messages*

**CPF3165**

Node group &1 in library &2 could not be created.
Create Node List (CRTNODL)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Create Node List (CRTNODL) command allows the user to create a node list object. This object is used to store node names that identify a set of systems in a network.

Note: Node lists can be used by system functions to indicate an operation is to be performed on a set of systems.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>NODL</td>
<td>Node list</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Node list</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>TEXT</td>
<td>Text ‘description’</td>
<td>Character value, *BLANK</td>
<td>Optional</td>
</tr>
</tbody>
</table>

Node list (NODL)

Specifies the qualified name of the node list object to be created.

The node list name can be qualified by one of the following library values:

*CURLIB
The node list is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name
Specify the name of the library where the node list is to be created.

The possible values are:

node-list-object-name
Specify the name of the node list to be created.

Text ’description’ (TEXT)

Specifies text that briefly describes the node list.

The possible values are:
*BLANK
Text is not specified.

'description'
Specify no more than 50 characters of text, enclosed in apostrophes.

Authority (AUT)
Specifies the authority given to users who do not have specific authority to the object, who are not on the authorization list, and whose user group has no specific authority to the object.

The possible values are:

*LIBCRTAUT
The public authority for the object is taken from the CRTAUT value of the target library (the library that is to contain the object). This value is determined when the object is created. If the CRTAUT value for the library changes after the object is created, the new value does not affect any existing objects.

*CHANGE
The user can perform all operations on the file except those limited to the owner or controlled by object existence authority and object management authority. The user can perform basic functions on the file, and the user can change it. Change authority provides object operational authority and all data authority.

*ALL
The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object’s existence, specify the security for the object, change the object, and perform basic functions on the object. The user can transfer ownership of the object.

*USE
The user can perform basic operations on the object, such as running a program or reading a file. The user cannot change the object. Use (*USE) authority provides object operational (*OBJOPR), read (*READ), and execute (*EXECUTE) authorities.

*EXCLUDE
The user is prevented from accessing the object.

authorization-list-name
Specify the name of the authorization list used.

Examples
Example 1: Node List Creation without Text Description
CRTNODL NODL(QGPL/NODL01)
This command creates a node list in library QGPL called NODL01. The node list has the same public authority as that defined for QGPL and it does not have a text description.

Example 2: Node List Creation with Text Description
CRTNODL NODL(MYLIB/NODL02) AUT(*EXCLUDE)
TEXT('This is my Node List number 2')
This command creates node list NODL02 in library MYLIB with public *EXCLUDE authority. The text description for this node list is 'This is my Node List number 2'.
Error messages

*ESCAPE Messages

CPF2108
Object &1 type *&3 not added to library &2.

CPF2112
Object &1 in &2 type *&3 already exists.

CPF2113
Cannot allocate library &1.

CPF2151
Operation failed for &2 in &1 type *&3.

CPF2182
Not authorized to library &1.

CPF2283
Authorization list &1 does not exist.

CPF9810
Library &1 not found.

CPF9820
Not authorized to use library &1.
Create NetBIOS Description (CRTNTBD)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Create NetBIOS Description (CRTNTBD) command creates a NetBIOS configuration description. Configuration objects, such as network server descriptions, for IOPs that support NetBIOS can then refer to this object for their NetBIOS parameters.

More information about using this command is in the Communications Configuration book, SC41-5401 book.

### Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>NTBD</td>
<td>NetBIOS description</td>
<td>Name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td>TEXT</td>
<td>Text ‘description’</td>
<td>Character value, *BLANK</td>
<td>Optional</td>
</tr>
<tr>
<td>FULLBFDTG</td>
<td>Full buffer datagrams</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
<tr>
<td>ADPWNDIV</td>
<td>Adaptive window interval</td>
<td>0-65535, 1000</td>
<td>Optional</td>
</tr>
<tr>
<td>MAXWDWERR</td>
<td>Maximum window errors</td>
<td>0-10, 0</td>
<td>Optional</td>
</tr>
<tr>
<td>MAXRCVDA</td>
<td>Maximum receive data size</td>
<td>512-16384, 4168</td>
<td>Optional</td>
</tr>
<tr>
<td>INACTTM</td>
<td>Inactivity timer</td>
<td>1000-65535, 30000</td>
<td>Optional</td>
</tr>
<tr>
<td>RSPTMR</td>
<td>Response timer</td>
<td>50-65535, 500</td>
<td>Optional</td>
</tr>
<tr>
<td>ACKTM</td>
<td>Acknowledgement timer</td>
<td>50-65535, 200</td>
<td>Optional</td>
</tr>
<tr>
<td>MAXIN</td>
<td>Maximum outstanding receives</td>
<td>1-127, 1</td>
<td>Optional</td>
</tr>
<tr>
<td>MAXOUT</td>
<td>Maximum outstanding transmits</td>
<td>1-127, 1</td>
<td>Optional</td>
</tr>
<tr>
<td>QRYTMR</td>
<td>Query timeout</td>
<td>500-10000, 500</td>
<td>Optional</td>
</tr>
<tr>
<td>NTBRTY</td>
<td>NetBIOS retry</td>
<td>1-50, 8</td>
<td>Optional</td>
</tr>
<tr>
<td>ALWMULTACK</td>
<td>Allow multiple acknowledgement</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
<tr>
<td>PREBLTPKT</td>
<td>Prebuilt message packets</td>
<td>1-200, 5</td>
<td>Optional</td>
</tr>
<tr>
<td>PKTRESTART</td>
<td>Packet confirms for restart</td>
<td>0-9999, 2</td>
<td>Optional</td>
</tr>
<tr>
<td>DLRTY</td>
<td>DLC retries</td>
<td>1-65535, 5</td>
<td>Optional</td>
</tr>
<tr>
<td>ETHSTD</td>
<td>Ethernet standard</td>
<td>*IEEE8023, *ETHV2</td>
<td>Optional</td>
</tr>
</tbody>
</table>
**NetBIOS description (NTBD)**

Specifies the name of the NetBIOS configuration object being created.

This is a required parameter.

---

**Text ’description’ (TEXT)**

Specifies text that briefly describes the NetBIOS description.

*BLANK

   Text is not specified.

‘description’

   Specify no more than 50 characters of text, enclosed in apostrophes.

---

**Full buffer datagrams (FULLBUFDTG)**

Specifies whether to request the full transmit buffer size for datagrams.

*NO

   The full transmit buffer size is not requested. The length of a datagram is equal to the transmit buffer size minus the size of the overhead, for a maximum of 512 bytes. Large messages are truncated.

   Note: The size of the overhead is the sum of the sizes of the NetBIOS header (44 bytes), the LAN header (a maximum of 36 bytes), and the buffer hold overhead (a maximum of 6 bytes).

*YES

   The full transmit buffer size is requested.

---

**Adaptive window interval (ADPWDWITV)**

Specifies the time, in milliseconds, between runs of the adaptive window algorithm. For each link, this algorithm is used to change the values on the MAXIN and MAXOUT parameters to match the values set on the remote workstation using NetBIOS protocol. The algorithm considers the conditions of the link, including adapter receive buffers and transmission load, when changing the values.

1000

   The time between runs of the adaptive window algorithm is 1000 milliseconds.

**adaptive-window-interval**

   Specify the time between algorithm runs, in milliseconds. Valid values range from 0 through 65535.

   Note: The value 0 disables the algorithm.

---

**Maximum window errors (MAXWDWERR)**

Specifies the number of dropped packets the adaptive window algorithm allows before decreasing the value on the MAXOUT parameter.

0

   The number of dropped packets is 0.
**window-errors**

Specify the number of window errors allowed. Valid values range from 0 through 10.

**Maximum receive data size (MAXRCVDATA)**

Specifies the maximum data size in any frame that can be received in a session. The partner in the transmission limits the size to the smaller of this specified size, or the size available in the partner’s transmit buffer. NetBIOS takes into account the maximum size that is forwarded by bridges in the path.

4168 The maximum data size that can be received is 4168 bytes.

**maximum-receive-data-size**

Specify the maximum data size that can be received, in bytes. Valid values range from 512 through 16384.

**Inactivity timer (INACTTMR)**

Specifies the amount of time that a link can be inactive before the NetBIOS protocol driver checks to verify that the link is operational.

30000 The link can be inactive for 30000 milliseconds.

**inactivity-timer**

Specify the amount of time to wait for activity, in milliseconds. Valid values range from 1000 through 65535.

**Response timer (RSPTMR)**

Specifies the amount of time to wait before again transmitting a link-level frame when no acknowledgement is received from the previous transmission.

500 The NetBIOS protocol driver waits 500 milliseconds.

**response-timer**

Specify the amount of time to wait, in milliseconds. Valid values range from 50 through 65535.

**Acknowledgement timer (ACKTMR)**

Specifies the amount of time the NetBIOS protocol driver delays acknowledging a received frame, when the number of frames sent is less than the maximum specified on the MAXIN parameter.

200 The driver delays for 200 milliseconds.

**acknowledgement-timer**

Specify the amount of time to delay, in milliseconds. Valid values range from 50 through 65535.
Maximum outstanding receives (MAXIN)

Specifies the maximum number of NetBIOS messages packets that can be received before sending an acknowledgement.

1 An acknowledgement is sent after one packet is received.

maximum-receives

Specify the number of packets to receive. Valid values range from 1 through 127.

Maximum outstanding transmits (MAXOUT)

Specifies the maximum number of NetBIOS messages packets that can be sent before expecting an acknowledgement.

Note: This parameter is used only when ADPWDWITV(0) is specified.

1 An acknowledgement is expected after one packet is sent.

maximum-transmits

Specify the number of packets to send. Valid values range from 1 through 127.

Query timeout (QRYTMR)

Specifies the time, in milliseconds, to wait between transmission retry attempts.

500 The time to wait is 500 milliseconds.

query-timeout

Specify a value in the range of 500 through 10000 milliseconds.

NetBIOS retry (NTBRTY)

Specifies the number of transmission retries that are attempted at the NetBIOS level before assuming that the receiving party is not present.

8 The number of retries is 8.

NetBIOS-retry

Specify a value in the range of 1 through 50 attempts.

Allow multiple acknowledgement (ALWMULTACK)

Specifies whether acknowledgements for received data can be combined with requests for data.

Note: When the NetBIOS protocol driver sends and receives acknowledgements with incoming data, LAN performance is improved.

*YES The acknowledgements can be combined with data requests.
Note: Both parties to the transmission must support combining acknowledgements with data requests or this value is ignored.

*NO The acknowledgements cannot be combined with data requests.

**Prebuilt message packets (PREBLTPKT)**

Specifies the number of NetBIOS message packets that are prebuilt for each session.

5 The number of NetBIOS message packets is 5.

*prebuilt-packets*

Specify a value in the range of 1 through 200 message packets.

**Packet confirms for restart (PKTRESTART)**

Specifies the number of transmission confirmations that must be received before sending additional packets when an out-of-resource condition occurs. The NetBIOS protocol driver stops sending packets when an out-of-resource condition is received from a port.

2 The maximum number of transmission confirmations is 2.

*packet-restart*

Specify the number of transmission confirmations. Valid values range from 0 through 9999.

**DLC retries (DLCRTY)**

Specifies the number of additional transmission attempts that will be made before assuming that the receiving data control link (DLC) layer is not responding.

5 The additional number of transmission attempts is 5.

*DLC-retries*

Specify a value in the range of 1 through 65535 attempts.

**Ethernet standard (ETHSTD)**

Specifies the Ethernet standard frame type that is used for NetBIOS communication.

*IEEE8023* IEEE 802.3 frames are used.

*ETHV2* Ethernet Version 2 frames are used.
Authority (AUT)

Specifies the authority you are giving to users who do not have specific authority for the object, who are not on an authorization list, and whose group profile or supplemental group profiles do not have specific authority for the object.

*CHANGE

The user can perform all operations on the object except those limited to the owner or controlled by object existence (*OBJEXIST) and object management (*OBJMGT) authorities. The user can change and perform basic functions on the object. *CHANGE authority provides object operational (*OBJOPR) authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove users.

*ALL

The user can perform all operations except those limited to the owner or controlled by authorization list management (*AUTLMGT) authority. The user can control the object’s existence, specify the security for the object, change the object, and perform basic functions on the object. The user also can change ownership of the object.

*USE

The user can perform basic operations on the object, such as running a program or reading a file. The user cannot change the object. Use (*USE) authority provides object operational (*OBJOPR), read (*READ), and execute (*EXECUTE) authorities.

*EXCLUDE

The user cannot access the object.

*LIBCRTAUT

The system determines the authority for the object by using the value specified for the Create authority (CRTAUT) parameter on the Create Library (CRTLIB) command for the library containing the object to be created. If the value specified for the CRTAUT parameter is changed, the new value will not affect any existing objects.

name

Specify the name of an authorization list to be used for authority to the object. Users included in the authorization list are granted authority to the object as specified in the list. The authorization list must exist when the object is created.

Examples

CRTNTBD NTBD(MYNETBIOS) ADPWDWITV(6000)

This command creates a NetBIOS description named MYNETBIOS specifying that the adapter window algorithm is to be run every 6000 milliseconds (6 seconds).

Error messages

*ESCAPE Messages

CPF26C3

IPX description &1 not created due to errors.

CPF27A6

NetBIOS description &1 not created due to errors.
Create NetWare Volume (CRTNTWVOL)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Create NetWare Volume (CRTNTWVOL) command creates a volume for the specified network server. The network server must be active at the time this command is run. The space is allocated from the device number.

This command causes the following to happen:
1. The physical volume is created on the specified device.
2. For NetWare 4+, the volume object, which represents the physical volume, is automatically placed into the NDS (NetWare Directory Services) tree. The volume object is put into the same container as the server object which represents the network server.
3. The volume is mounted if specified.

The following restrictions apply for NetWare volumes:
- A maximum of 8 segments per device is allowed. A device can have a maximum of 8 segments from the same volume or combination of different volumes.
- A maximum of 32 segments per volume is allowed. CRTNTWVOL will create the first segment. The Change NetWare Volume (CHGNTWVOL) command can be used, up to 31 times, to add additional segments.
- A maximum of 64 mounted volumes per server is allowed.

Restrictions: You must have *IOSYSCFG special authority to use this command.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>Vol</td>
<td>Volume</td>
<td>Character value</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td>Server</td>
<td>Server</td>
<td>Character value</td>
<td>Required, Positional 2</td>
</tr>
<tr>
<td>Size</td>
<td>Size of volume</td>
<td>1-8000, *MAXAVAIL</td>
<td>Optional</td>
</tr>
<tr>
<td>Devnbr</td>
<td>Device number</td>
<td>Character value</td>
<td>Optional</td>
</tr>
<tr>
<td>Segnbr</td>
<td>Segment number</td>
<td>0-14, *ANY</td>
<td>Optional</td>
</tr>
<tr>
<td>Blksize</td>
<td>Block size</td>
<td>*DFT, 4, 8, 16, 32, 64</td>
<td>Optional</td>
</tr>
<tr>
<td>Dtacpr</td>
<td>Data compression</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
<tr>
<td>Blksubalc</td>
<td>Block suballocation</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
<tr>
<td>Mount</td>
<td>Mount</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
<tr>
<td>Text</td>
<td>Text</td>
<td>Character value, *Vol</td>
<td>Optional</td>
</tr>
</tbody>
</table>

© Copyright IBM Corp. 1998, 2006
Volume (VOL)
Specifies the physical volume to be created. By default, the name of the volume object placed into the NDS tree is servername_volumename. The volume name must be 2 - 15 characters long and must be unique within the directory tree.

Server (SERVER)
Specifies the server for which the volume is being created. This network server must be active when this command is executed.

Size of volume (SIZE)
Specifies the size of the volume to be created, in megabytes.
*MAXAVAIL
Use the size of the largest area of contiguous free space on the specified device.
1-8000 Specify the size of the volume in megabytes. The size specified must be no larger than the largest area of contiguous free space on the device.

Device number (DEVNBR)
Specifies the device number from which the storage for the volume is allocated. A value of 0 to 99 is allowed to be specified. Use Work with Netware Volumes (WRKNTWVOL) command to find an available device number.

Segment number (SEGNBR)
Specifies the number of the segment which the new volume is allocated.
*ANY Use the first segment which has the available size requested.
0-14 Specify the number of the segment from which the storage for the new volume is to be allocated. Use WRKNTWVOL (F10=Show Segments) to determine which segments are free and available for use.

Block size (BLKSIZE)
Specifies the block size of the data in the volume. This value cannot be changed once the volume is created. Volume blocks are the smallest piece of data the server can access in a volume. The larger the block size, the greater the potential for wasted memory. However, if the majority of files are very large, a large block size may be more efficient. Turning on block suballocation will aid in minimizing waste. Block suballocation is controlled by the Block suballocation (BLKSUBALC) parameter.

*DFT Specifies to use the block size that NetWare determines by default given the size of the volume.
**block-size**
Specify the block size of the data, in kilobytes. Valid values are: 4, 8, 16, 32 and 64 kilobytes.

---

**Data compression (DTACPR)**
Specifies whether or not to enable data compression for this volume. Note that once data compression is turned on for a volume, it cannot be turned off.

* **YES** Data compression is enabled for this volume.
* **NO** Data compression is not enabled for this volume.

---

**Block suballocation (BLKSUBALC)**
Specifies whether or not to enable block suballocation for this volume. Note that once block suballocation is turned on for a volume, it cannot be turned off.

* **YES** Block suballocation is enabled for this volume.
* **NO** Block suballocation is not enabled for this volume.

---

**Mount (MOUNT)**
Specifies whether or not to mount this volume as soon as it’s created. Don’t mount volumes that are rarely used because each mounted volume uses some server memory. Volumes can be mounted and dismounted while the server is active.

* **YES** The volume is created and then mounted.
* **NO** The volume is created but not mounted.

---

**Text (TEXT)**
Specifies the text that briefly describes the volume.

* **VOL** The volume name will be used for the description of the volume.

**character-value**
Specify no more than 50 characters of text, enclosed in apostrophes.

---

**Examples**

**Example 1: Creating a Physical Volume**

CRTNTWVOL VOL(APPS) SERVER(SERVER1) NWSSTG(STGSPACE1)

This command creates physical volume APPS for network server SERVER1 with the size of the largest contiguous free space available on the network server storage space. SERVER1 is a local server (a NWSD
of type *NETWARE defined on the local system). The storage is allocated from storage space STGSPACE1. A volume object called SERVER1_APPS is placed into the NDS tree. The volume is mounted by default.

**Example 2: Creating a Volume with Specific Size**

```plaintext
CRTNTWVL VOL(VOL1) SERVER(SERVER2) DEVNBR(9)
  SIZE(50) DTACPR(*YES) MOUNT(*NO)
```

This command creates physical volume VOL1 for network server SERVER2 with a size of 50 megabytes. This server may be local or remote. The storage is allocated from device number 9. A volume object called SERVER2_VOL1 is placed into the NDS tree. Data compression is enabled for this volume and the volume is not mounted.

**Error messages**

**ESCAPE Messages**

FPE0107
  Volume &1 not created.

---

36  IBM Systems - iSeries: i5/OS Commands Starting with CRTMSGF (Create Message File)
The Create Network Interface (Frame-Relay Network) (CRTNWIFR) command creates a network interface for a frame-relay (FR) network. More information about using this command is in the Communications Configuration book, SC41-5401.

**Parameters**

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>NWID</td>
<td>Network interface description</td>
<td>Name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td>RSRCNAME</td>
<td>Resource name</td>
<td>Name</td>
<td>Required, Positional 2</td>
</tr>
<tr>
<td>ONLINE</td>
<td>Online at IPL</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
<tr>
<td>VRYWAIT</td>
<td>Vary on wait</td>
<td>15-180, *NOWAIT</td>
<td>Optional</td>
</tr>
<tr>
<td>DLCI</td>
<td>Data link connection ID</td>
<td>Single values: *NONE Other values (up to 256 repetitions): Element list</td>
<td>Optional</td>
</tr>
<tr>
<td>NRZI</td>
<td>NRZI data encoding</td>
<td>*NO, *YES</td>
<td>Optional</td>
</tr>
<tr>
<td>INTERFACE</td>
<td>Physical interface</td>
<td>*RS449V36, *V35, *X21</td>
<td>Optional</td>
</tr>
<tr>
<td>CLOCK</td>
<td>Clocking</td>
<td>*MODEM, *LOOP, *INVERT</td>
<td>Optional</td>
</tr>
<tr>
<td>LINESPEED</td>
<td>Line speed</td>
<td>56000-2048000, 56000, 64000, 128000, 192000, 256000, 320000, 384000, 448000, 512000, 1024000, 1536000, 2048000</td>
<td>Optional</td>
</tr>
<tr>
<td>LLMIMODE</td>
<td>LMI mode</td>
<td>*TE, *FH, *NONE, *ANNEXA</td>
<td>Optional</td>
</tr>
<tr>
<td>POLLITV</td>
<td>Polling interval</td>
<td>5-30, 10</td>
<td>Optional</td>
</tr>
<tr>
<td>FULLINQTV</td>
<td>Full inquiry interval</td>
<td>1-255, 6</td>
<td>Optional</td>
</tr>
<tr>
<td>TEXT</td>
<td>Text ‘description’</td>
<td>Character value, *BLANK</td>
<td>Optional</td>
</tr>
<tr>
<td>CMNRCYLMT</td>
<td>Recovery limits</td>
<td>Single values: *SYSVAL Other values: Element list</td>
<td>Optional</td>
</tr>
</tbody>
</table>

**Network interface description (NWID)**

This is a required parameter.

Specifies the name of the network interface description.
**network-interface-description-name**

Specify the name of a network interface description.

---

**Resource name (RSRCNAME)**

This is a required parameter.

Specifies the resource name that identifies the hardware that the description represents.

**Note:** Use the Work with Hardware Resources (WRKHDWRSC) command with *CMN specified for the TYPE parameter to help determine the resource name. Specify the resource name of the communications port. The resource name consists of the input/output adapter (IOA) resource name and the port number on the IOA. For example, if the resource name of the IOA is LIN01 and the port on the IOA is 1, then the resource name is LIN011.

**resource-name**

Specify a resource name.

---

**Online at IPL (ONLINE)**

Specifies whether this object is automatically varied on at initial program load (IPL).

*YES The network interface is automatically varied on at initial program load (IPL).

*NO This network interface is not automatically varied on at IPL.

---

**Vary on wait (VRYWAIT)**

Specifies whether the network interface is varied on asynchronously or synchronously. For synchronous vary on, specifies how long the system waits for the vary on to complete.

*NOWAIT

The system does not wait for the vary on to complete. The network interface is varied on asynchronously.

**vary-on-wait**

Specify the time (in seconds) to wait. Valid values range from 15 through 180. The system waits until the network interface is varied on, or until the specified time passes, before completing the Vary Configuration (VRYCFG) command.

**Notes:**

1. Specifying a wait time in the network interface description affects system IPL time, if ONLINE(*YES) is used, by the amount of time it takes to synchronously vary on the network interface or reach the wait-time value.

2. The time required to vary on a network interface is the time it takes to put tasks in place to manage the network interface, to activate the communications I/O processor (IOP) (including downloading the IOP model-unique Licensed Internal Code), and to establish communications with the data circuit-terminating equipment (DCE). Normal vary-on time ranges from 5 through 45 seconds, but can be longer, depending on the system, network interface protocol, and other factors.
Data link connection ID (DLCI)

Specifies the data link connection identifiers of the line descriptions for the frame relay network interface being created.

*NONE

No data link connection identifier is specified.

Element 1: DLCI Number

DLCI-number

Specify the DLCI number for the line.

Element 2: Line Description

line-description

Specify the DLCI line description.

NRZI data encoding (NRZI)

Specifies whether non-return-to-zero-inverted (NRZI) data encoding is used for modems that are sensitive to certain bit patterns in the data stream. This ensures that the signal does not stay the same for an extended period of time.

Note: All data communications equipment on the line must use the same transmission method.

*NO NRZI data encoding is not used.
*YES NRZI data encoding is used.

Physical interface (INTERFACE)

Specifies the type of physical interface on the input/output adapter (IOA) port.

*RS449V36

An RS-499/V.36 physical interface is used. This value is valid only for frame relay and SDLC links.

*V35 A V.35 physical interface is used. This value is valid only for frame relay, BSC, and SDLC links.

*X21 An X.21 physical interface is used. This value is valid only for frame relay, X.25, and SDLC links.

Clocking (CLOCK)

Specifies the method in which the clocking function is provided for the network interface.

*MODEM

The modem provides the clocking.
*LOOP
The system inverts the clock from the modem and uses it as the transmit clock on the line.

*INVERT
The transmit clock provided by the modem data circuit-terminating equipment (DCE) is inverted before use. This option can be used when having problems with high speed data transmission and the modem (DCE) does not support looped clocking.

**Line speed (LINESPEED)**
Specifies the line speed in bits per second (bps)

<table>
<thead>
<tr>
<th>Value</th>
<th>Line Speed</th>
</tr>
</thead>
<tbody>
<tr>
<td>1536000</td>
<td>The line speed is 1536000 bps.</td>
</tr>
<tr>
<td>56000</td>
<td>The line speed is 56000 bps.</td>
</tr>
<tr>
<td>64000</td>
<td>The line speed is 64000 bps.</td>
</tr>
<tr>
<td>128000</td>
<td>The line speed is 128000 bps.</td>
</tr>
<tr>
<td>192000</td>
<td>The line speed is 192000 bps.</td>
</tr>
<tr>
<td>256000</td>
<td>The line speed is 256000 bps.</td>
</tr>
<tr>
<td>320000</td>
<td>The line speed is 320000 bps.</td>
</tr>
<tr>
<td>384000</td>
<td>The line speed is 384000 bps.</td>
</tr>
<tr>
<td>448000</td>
<td>The line speed is 448000 bps.</td>
</tr>
<tr>
<td>512000</td>
<td>The line speed is 512000 bps.</td>
</tr>
<tr>
<td>1024000</td>
<td>The line speed is 1024000 bps.</td>
</tr>
<tr>
<td>1536000</td>
<td>The line speed is 1536000 bps.</td>
</tr>
<tr>
<td>2048000</td>
<td>The line speed is 2048000 bps.</td>
</tr>
</tbody>
</table>

The line speed is specified in bits per second (bps). Valid values range from 56000 bps through 2048000 bps.

**LMI mode (LMIMODE)**
Specifies whether the local management interface (LMI) for this adapter is configured as terminal equipment or a frame handler.

*TE
The local system is configured to interface with a frame relay network as terminal equipment. The frame relay network must be set to operate at ANSI T1.617 Annex D, to be compatible with system link management frames.

*FH
The local system is configured to interface with another system as a frame handler. In this configuration, the local system is performing as the frame relay network.

*ANNEXA
The local system is configured to interface with a frame relay network as terminal equipment.
The frame relay network must operate as an ITU (previously CCITT) Q.933 Annex A to be compatible with system link management frames.

*NONE
The local system is configured to interface with the frame relay network or another system without performing any LMI function.

Polling interval (POLLITV)
Specifies the rate of the polling cycle. The polling cycle consists of a status inquiry message and a status message exchange. The status message includes the status of the DLCI.

10 A polling interval of 10 seconds is used.

polling-interval
Specify the polling interval to be used. Valid values range from 5 through 30 seconds.

Full inquiry interval (FULLINQITV)
Specifies the number of polling cycles that occur before a full status inquiry is requested.

6 A full inquiry interval of 6 polling cycles is used.

full-inquiry-interval
Specify the number of polling cycles for a full status cycle to be requested. Valid values range from 1 through 255.

Text 'description' (TEXT)
Specifies text that briefly describes the network interface.

*BLANK Text is not specified.

character-value Specify no more than 50 characters of text, enclosed in apostrophes.

Recovery limits (CMNRCYLMNT)
Specifies the number of recovery attempts made by the system before an inquiry message is sent to the system operator. Also specifies the time (in minutes) that must elapse before the system sends an inquiry message to the system operator indicating that the recovery attempt count limit is reached.

Element 1: Maximum Recovery Limit

2 Two recovery attempts are made within the interval specified.

count-limit Specify the number of recovery attempts to be made. Valid values range from 0 through 99.
Element 2: Recovery Time Interval

5 A 5-second time-out period is used.

time-interval

Specify the time interval (in minutes) at which the specified number of second-level recoveries are attempted. Valid values range from 0 through 120. If the value specified for count-limit is not 0, the value 0 specifies infinite recovery.

Other Single Value

*SYSVAL

The recovery limits specified in the QCMNRNCYLMT system value are used.

Authority (AUT)

Specifies the authority you are giving to users who do not have specific authority for the object, who are not on an authorization list, and whose group profile or supplemental group profiles do not have specific authority for the object.

*CHANGE

The user can perform all operations on the object except those limited to the owner or controlled by object existence (*OBJEXIST) and object management (*OBJMGT) authorities. The user can change and perform basic functions on the object. *CHANGE authority provides object operational (*OBJOPR) authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove users.

*ALL

The user can perform all operations except those limited to the owner or controlled by authorization list management (*AUTLMGT) authority. The user can control the object’s existence, specify the security for the object, change the object, and perform basic functions on the object. The user also can change ownership of the object.

*USE

The user can perform basic operations on the object, such as running a program or reading a file. The user cannot change the object. Use (*USE) authority provides object operational (*OBJOPR), read (*READ), and execute (*EXECUTE) authorities.

*EXCLUDE

The user cannot access the object.

*LIBCRAFT

The system determines the authority for the object by using the value specified for the Create authority (CRTAUT) parameter on the Create Library (CRTLIB) command for the library containing the object to be created. If the value specified for the CRTAUT parameter is changed, the new value will not affect any existing objects.

name Specify the name of an authorization list to be used for authority to the object. Users included in the authorization list are granted authority to the object as specified in the list. The authorization list must exist when the object is created.

Examples

CRTNWIFR NWID((THISONE)) RSRCHNAME(LINE031) ONLINE(+YES) VRWYWAIT(15) DLCI((32 LINEABC) (409 LINEDEF) (94 LINELAST)) INTERFACE(+V35) LMIMODE(*TE)
This command creates the frame relay network interface THISONE. THISONE represents the resource named LINE031. THISONE is varied on at initial program load (IPL) with a vary on wait time of 15 seconds. It is created with three DLCIs (32, 409, and 94) which refer to line descriptions LINEABC, LINEDEF, and LINELAST respectively. The type of physical interface for the input/output adapter (IOA) port specified by THISONE is *V35. The local management interface mode is configured to interface with a frame relay network as terminal equipment (TE).

**Error messages**

*ESCAPE Messages*

**CPF27A0**

Network interface description &1 not created due to errors.
Create NWS Configuration (CRTNWSCFG)

Where allowed to run: All environments (*ALL)
Threads: No

The Create NWS Configuration (CRTNWSCFG) command creates a network server configuration and an associated validation list. The network server configuration defines connection security or remote system attributes.

Restrictions:
- This command is shipped with public exclude (*EXCLUDE) authority. When this command is shipped, authority is issued only to the security officer. The security officer can grant the use of this command to other users.
- You must have input/output system configuration (*IOSYSCFG) special authority to use this command.
- To specify a non-default value for the IPSECRULE, CHAPAUT, or SPCERTID parameters, you must have security administrator (*SECADM) special authority.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>NWSCFG</td>
<td>Network server configuration</td>
<td>Communications name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td>TYPE</td>
<td>Configuration type</td>
<td>*CNNSEC, *RMTSYS, *SRVPRC</td>
<td>Required, Positional 2</td>
</tr>
<tr>
<td>IPSECRULE</td>
<td>IP security rules</td>
<td>Single values: *NONE Other values (up to 16 repetitions): Character value, *GEN, *REGEN</td>
<td>Optional</td>
</tr>
<tr>
<td>INZSP</td>
<td>Initialize service processor</td>
<td>*MANUAL, *AUTO, *NONE</td>
<td>Optional</td>
</tr>
<tr>
<td>ENBUNICAST</td>
<td>Enable unicast</td>
<td>*NO, *YES</td>
<td>Optional</td>
</tr>
<tr>
<td>SPNAME</td>
<td>Service processor name</td>
<td>Character value, *SPINTNETA</td>
<td>Optional</td>
</tr>
<tr>
<td>SPINTNETA</td>
<td>SP internet address</td>
<td>Character value</td>
<td>Optional</td>
</tr>
<tr>
<td>SPCERTID</td>
<td>SP certificate identifier</td>
<td>Single values: *NONE Other values: *COMMONNAME, *EMAIL, *ORGUNIT</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Element 1: Component</td>
<td>*COMMONNAME, *EMAIL, *ORGUNIT</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Compare value</td>
<td>Character value</td>
<td></td>
</tr>
<tr>
<td>EID</td>
<td>Enclosure identifier</td>
<td>Single values: *AUTO Other values: *COMMONNAME, *EMAIL, *ORGUNIT</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Element 1: Serial number</td>
<td>Character value</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Manufacturer type and model</td>
<td>Character value</td>
<td></td>
</tr>
<tr>
<td>SPNWSCFG</td>
<td>SP configuration name</td>
<td>Communications name</td>
<td>Optional</td>
</tr>
<tr>
<td>RMTSYSID</td>
<td>Remote system identifier</td>
<td>Single values: *SPNWSCFG Other values: *COMMONNAME, *EMAIL, *ORGUNIT</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Element 1: Serial number</td>
<td>Character value</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Manufacturer type and model</td>
<td>Character value</td>
<td></td>
</tr>
</tbody>
</table>
### Network server configuration (NWSCFG)

Specifies the name of the network server configuration.

This is a required parameter.

**name**  Specify the name of the network server configuration to be created.

### Configuration type (TYPE)

Specifies the type of network server configuration to be created.

This is a required parameter.

*CNNSEC  
Connection security defines the IP Security (IPSec) rule attributes.
*RMTSYS
Remote system defines the hardware and configuration attributes required to boot the server.

*SRVPRC
Service processor defines the attributes used to locate and manage the server.

---

**IP security rules (IPSECRULE)**

Specifies the configuration IP Security (IPSec) rules used between the hosting and remote system.

This parameter is only valid when TYPE(*CNNSEC) is specified.

**Single values**

*NONE
IP Security (IPSec) protocol security settings are not configured.

**Other values (up to 16 repetitions)**

*GEN Generate a random pre-shared key.

*REGEN
Automatically generate a random pre-shared key every time the system is varied on.

**character-string**
Specify the pre-shared key.

A pre-shared key is a nontrivial string up to 32 characters long.

Valid characters are upper case A through Z, lower case a through z, numbers 0 through 9, and the following special characters:

- Plus sign
- Equal sign
- Percent
- Ampersand
- Left parenthesis
- Right parenthesis
- Comma
- Underline
- Minus sign
- Period
- Colon
- Semicolon

---

**Initialize service processor (INZSP)**

Specifies how the remote system’s service processor is secured.

This parameter is only valid when TYPE(*SRVPRC) is specified.

*MANUAL
To use this option, it is required that the remote system’s service processor is pre-configured with a user name, password and certificate. Certificate management will be required. This method is appropriate when connecting to the service processor via public networks to protect the password.

*AUTO

Parameters are automatically configured on the remote system’s service processor.

*AUTO provides security without requiring pre-configuration of the remote system’s service processor. The remote system’s service processor will have certificates automatically regenerated when the certificates are near expiration. This option is appropriate if the interconnecting network is physically secure or is protected by a firewall.

Note: An administrator will need to regenerate the certificate using the Initialize NWS Configuration (INZNWSCFG) command when the service processor certificate has expired, or if a new certificate and password are desired at any time before the certificate expires.

*NONE

Provides no security.

Use this only if the interconnecting network is physically secure.

Note: Some service processors do not support secure connections. Use *NONE for these service processors. Additional information can be found at Integrated xSeries solutions at http://www.ibm.com/servers/eserver/iseries/integratedxseries.

Enable unicast (ENBUNICAST)

Specifies whether unicast packet distribution is to be used. Unicast is a transmission method where packets are sent directly to the specified Service processor name (SPNAME) or SP internet address (SPINTNETA) parameter.

The system identification for the Enclosure identifier (EID) parameter is automatically retrieved if *AUTO is specified and the system hardware supports it.

This parameter is only valid when TYPE(*SRVPRC) is specified.

*NO Disable unicast

*YES Enable unicast.

Service processor name (SPNAME)

Specifies the remote system’s service processor host name.

This parameter is only valid when TYPE(*SRVPRC) is specified.

Note: This parameter is required when ENBUNICAST(*YES) is specified.

*SPINTNETA

The remote system is identified by the value specified for the SP internet address (SPINTNETA) parameter.
host-name
Specify the remote system’s service processor host name.

---

**SP internet address (SPINTNETA)**

Specifies the remote system’s service processor internet address.

This parameter is only valid when TYPE(*SRVPRC) is specified.

**Notes:**
1. This parameter is ignored when ENBUNICAST(*NO) is specified.
2. This parameter is required when SPNAME(*SPINTNETA) is specified.

internet-address
Specify the internet address of the service processor.

The value is entered in the decimal form _nnn.nnn.nnn.nnn_, where _nnn_ is a decimal number ranging from 0 through 255.

---

**SP certificate identifier (SPCERTID)**

The SP certificate identifier specifies one of three possible fields that identifies the service processor’s certificate.

This parameter is specified to provide additional validation that the certificate is from the service processor. The contents of the selected field must exactly match the value of the field that was entered when the certificate was generated or requested from a certificate authority.

This parameter is only valid when TYPE(*SRVPRC) is specified.

This parameter is required when INZSP(*MANUAL) is specified and cannot have the value *NONE.

**Single values**

*NONE
Service processor certificate is not configured.

**Element 1: Component**

*COMMONNAME
Selects the certificate’s common name specified when the certificate was generated or requested from a certificate authority. On the remote supervisor adapter II this correlates to the “ASM Domain Name” field used to generate a self-signed certificate or generate a certificate signing request.

*EMAIL
Selects the certificate’s e-mail address specified when the certificate was generated or requested from a certificate authority. On the remote supervisor adapter II this correlates to the “Email Address” field used to generate a self-signed certificate or generate a certificate signing request.

*ORGUNIT
Selects the certificate’s organizational unit specified when the certificate was generated or...
requested from a certificate authority. On the remote supervisor adapter II this correlates to the "Organizational Unit" field used to generate a self-signed certificate or generate a certificate signing request.

Element 2: Compare value

*character-string
Specify the certificates component compare value. Enter no more than 255 characters of text, enclosed in apostrophes.

---

**Enclosure identifier (EID)**

Specifies the identifying serial number, type and model of the enclosure containing the service processor. When specified, they are used to locate the system on the network.

Look for these values on the label of the system.

This parameter is only valid when TYPE(*SRVPRC) is specified.

**Single values**

*AUTO
Automatically retrieve the identifier when ENBUNICAST(*YES) is specified.

Element 1: Serial number

*character-string
Specify the machine serial number.

Element 2: Manufacturer type and model

*character-string
Specify the machine type and model.

The value is entered in the form ttttmmm where ttt is the machine type and mmm is the machine model number.

---

**SP configuration name (SPNWSCFG)**

Specifies the name of the service processor network server configuration to be used to manage the remote server.

This parameter is only valid when TYPE(*RMTSYS) is specified.

*name
Specifies the name of the service processor network server configuration.
**Remote system identifier (RMTSYSID)**

Specifies the identifying serial number, type and model of the remote system. When specified, they are used to locate the remote system on the network.

Look for these values on the label of the system.

**Note:** The machine type and model may be omitted if the system’s serial number is unique on the network.

This parameter is only valid when TYPE(*RMTSYS) is specified.

**Single values**

*SPNWSCFG

Use the serial number and type/model specified in the Enclosure identifier (EID) parameter of the service processor (*SRVPRC) network server configuration.

**Element 1: Serial number**

`character-string`

Specify the machine serial number.

**Element 2: Manufacturer type and model**

`character-string`

Specify the machine type and model.

The value is entered in the form `tttmmmm` where `ttt` is the machine type and `mmmm` is the machine model number.

**Delivery method (DELIVERY)**

Specifies how the parameters necessary to configure the remote system are delivered.

This parameter is only valid when TYPE(*RMTSYS) is specified.

*DYNAMIC

Parameters are dynamically delivered to the remote system using Dynamic Host Configuration Protocol (DHCP).

*MANUAL

Parameters are manually configured on the remote system using the BIOS utilities (System BIOS or Adapter BIOS - CTRL-Q).

**CHAP authentication (CHAPAUT)**

Specifies the Challenge Handshake Authentication Protocol (CHAP) for the host system iSCSI target to authenticate the remote system initiator node.

This parameter is only valid when TYPE(*RMTSYS) is specified.

**Single values**
*NONE
CHAP authentication is not enabled.

Element 1: CHAP name

*NWSCFG
The system will automatically generate a name for CHAP using the Network server configuration name.

*character-string
Specify the name you want to use for the Challenge Handshake Authentication Protocol.

Valid characters are upper case A through Z, lower case a through z, numbers 0 through 9, and the following special characters:
- Plus sign
- Equal sign
- Percent
- Ampersand
- Left parenthesis
- Right parenthesis
- Comma
- Underline
- Minus sign
- Period
- Colon
- Semicolon

Element 2: CHAP secret

*GEN
The system will automatically generate a random CHAP secret.

*character-string
Specify the secret you want to use for the Challenge Handshake Authentication Protocol.

Valid characters are upper case A through Z, lower case a through z, numbers 0 through 9, and the following special characters:
- Plus sign
- Equal sign
- Percent
- Ampersand
- Left parenthesis
- Right parenthesis
- Comma
- Underline
- Minus sign
- Period
- Colon
- Semicolon
Boot device ID (BOOTDEVID)

Specifies the PCI Function Address (Bus/Device/Function) of the iSCSI adapter in the remote system that will be used to boot from.

**Note:** Remote systems with more than one iSCSI adapter installed in the server are required to specify which adapter will be used to boot from.

This parameter is only valid when TYPE(*RMTSYS) is specified.

**Single values**

*SINGLE

The single iSCSI adapter is used on the remote system

**Element 1: Bus**

*number

Specify the bus number of the remote system’s iSCSI adapter that will be used to boot.

Valid values range from 0 through 255.

**Element 2: Device**

*number

Specify the device number of the remote system’s iSCSI adapter that will be used to boot.

Valid values range from 0 through 31.

**Element 3: Function**

*number

Specify the function number of the remote system’s iSCSI adapter that will be used to boot.

Valid values range from 0 through 7.

Dynamic boot options (DYNBOOTOPT)

Specifies the internal Dynamic Host Configuration Protocol (DHCP) Server configuration.

**Note:** This is an advanced configuration function.

This parameter is used to configure the internal DHCP Server that is part of the iSCSI Target Host Bus Adapter firmware. It is used to provide IP address and diskless boot parameters for the remote iSCSI Initiator.

This parameter is only valid when TYPE(*RMTSYS) is specified.

This parameter is only valid when DELIVERY(*DYNAMIC) is specified.

**Element 1: Vendor ID**

The client and server are pre-configured to a default vendor ID. Network administrators can configure clients to define their own identifying values to convey hardware, operating system or other identifying information. DHCP option 60 described in the IETF RFC 2132 is used for this function.

*DFT The default vendor ID will be used.
character-string
Vendor ID of the remote system’s iSCSI adapter that will be used.

Element 2: Alternate client ID

Used by clients to specify their unique identifier to the server. Each client’s identifier must be unique among all other client identifiers used on the effective DHCP network to which the client is attached (that is, the client’s local subnet and any remote subnets reachable using DHCP relay). Vendors and system administrators are responsible for choosing client identifiers that meet this requirement for uniqueness. DHCP option 61 described in the IETF RFC 2132 is used for this function.

*ADPT
The default Client ID consists of the adapter address for the remote system’s iSCSI adapter. This value will be used to identify the remote system.

character-string
Specify the Client ID of the remote system’s iSCSI adapter that will be used to boot.

Remote interfaces (RMTIFC)

Specifies the remote system’s interfaces. This information is used to identify and configure the remote system’s interfaces. Each adapter has two functions to support a SCSI and a LAN interface.

This parameter is only valid when TYPE(*RMTSYS) is specified.

Note: A minimum of one SCSI interface and one LAN interface is required though they may reside on different adapters in the remote system.

You can specify up to 4 repetitions for this parameter.

Element 1: SCSI interface

Specifies the remote system’s SCSI interfaces.

Element 1: Adapter address

*NONE
No SCSI interface is configured for this adapter.

adapter-address
Specify the 12-character hexadecimal adapter address for the remote system’s iSCSI interface.

Element 2: Internet address

internet-address
Specify the internet address for the remote system’s SCSI interface.

The value is entered in the decimal form nnn.nnn.nnn.nnn, where nnn is a decimal number ranging from 0 through 255.

Element 3: Subnet mask

subnet-mask
Specify the subnet mask for the remote system’s SCSI interface.
The value is entered in the decimal form \textit{n\textsubscript{nnn}.n\textsubscript{nnn}.n\textsubscript{nnn}.n\textsubscript{nnn}} , where \textit{nnn} is a decimal number ranging from 0 through 255.

Element 4: Gateway address

\textit{gateway-address}

Specify the gateway address for the remote system’s SCSI interface.

The value is entered in the decimal form \textit{n\textsubscript{nnn}.n\textsubscript{nnn}.n\textsubscript{nnn}.n\textsubscript{nnn}} , where \textit{nnn} is a decimal number ranging from 0 through 255.

Element 5: iSCSI qualified name

\textbf{\textasteriskcentered GEN} The system will automatically generate the iSCSI qualified name.

\textit{name}

Specify the iSCSI qualified name for the remote system’s SCSI interface.

The following characters are allowed in iSCSI qualified names:

\begin{itemize}
  \item Alphabetical characters A through Z converted to lower case (refer to RFC 3722)
  \item Alphabetical characters a through z
  \item Digits 0 through 9
  \item Period (.)
  \item Dash (-)
  \item colon (:)
\end{itemize}

Element 2: LAN interface

Specifies the remote system’s LAN interfaces.

Element 1: Adapter address

\textbf{\textasteriskcentered NONE} No LAN interface is configured for this adapter.

\textit{adapter-address}

Specify the 12-character hexadecimal adapter address for the remote system’s LAN or TCP Offload Engine (TOE) interface.

Element 2: Internet address

\textit{internet-address}

Specify the internet address for the remote system’s LAN interface.

The value is entered in the decimal form \textit{n\textsubscript{nnn}.n\textsubscript{nnn}.n\textsubscript{nnn}.n\textsubscript{nnn}} , where \textit{nnn} is a decimal number ranging from 0 through 255.

Element 3: Subnet mask

\textit{subnet-mask}

Specify the subnet mask for the remote system’s LAN interface.

The value is entered in the decimal form \textit{n\textsubscript{nnn}.n\textsubscript{nnn}.n\textsubscript{nnn}.n\textsubscript{nnn}} , where \textit{nnn} is a decimal number ranging from 0 through 255.
Element 4: Gateway address

*gateway-address*

Specify the gateway address for the remote system’s LAN interface.

The value is entered in the decimal form nnn.nnn.nnn.nnn, where nnn is a decimal number ranging from 0 through 255.

---

Text 'description' (TEXT)

Specifies text that briefly describes the network server configuration.

*BLANK

Text is not specified.

*character-value*

Specify no more than 50 characters of text, enclosed in apostrophes.

---

Authority (AUT)

Specifies the authority you are giving to users who do not have specific authority for the object, who are not on an authorization list, and whose group profile or supplemental group profiles do not have specific authority for the object.

*CHANGE

The user can perform all operations on the object except those limited to the owner or controlled by object existence (*OBJEXIST) and object management (*OBJMGT) authorities. The user can change and perform basic functions on the object. *CHANGE authority provides object operational (*OBJOPR) authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove users.

*ALL

The user can perform all operations except those limited to the owner or controlled by authorization list management (*AUTLMGT) authority. The user can control the object’s existence, specify the security for the object, change the object, and perform basic functions on the object. The user also can change ownership of the object.

*USE

The user can perform basic operations on the object, such as running a program or reading a file. The user cannot change the object. Use (*USE) authority provides object operational (*OBJOPR), read (*READ), and execute (*EXECUTE) authorities.

*EXCLUDE

The user cannot access the object.

*LIBCRTAUT

The system determines the authority for the object by using the value specified for the Create authority (CRTCRTAUT) parameter on the Create Library (CRTLIB) command for the library containing the object to be created. If the value specified for the CRTAUT parameter is changed, the new value will not affect any existing objects.

*name*

Specify the name of an authorization list to be used for authority to the object. Users included in the authorization list are granted authority to the object as specified in the list. The authorization list must exist when the object is created.
Examples

Example 1: Security Connection Network Server Configuration
CRTNWSCFG NWSCFG(MYSEC)
  TYPE(*CNNSEC)
  IPSECRULE(*NONE)
  TEXT('Connection security')

This command creates a Connection Security Configuration with no security rules.

Example 2: Service Processor Network Server Configuration
CRTNWSCFG NWSCFG(CAT4SP)
  TYPE(*SRVPRC)
  INZSP(*MANUAL)
  ENBUNICAST(*YES)
  SPINTNETA('1.5.10.75')
  SPCERTID(*ORGUNIT 'ACME Corp')
  EID(RTYM14A 3305R8U)
  TEXT('CAT4SP Service processor')

This command creates a service processor configuration. The service processor is contacted using IP Address 1.5.10.75. The remote system is identified by the serial number RTYM14A. The service processor is manually secured using a certificate configured with the 'ACME Corp' organizational unit.

Example 3: Remote System Network Server Configuration
CRTNWSCFG NWSCFG(CAT4SVR)
  TYPE(*RMTSYS)
  SPNWSCFG(CAT4SP)
  RMTSYSID(*SPNWSCFG)
  DELIVERY(*DYNAMIC)
  RMTIFC(((111111111111 '1.5.10.10' '255.255.255.0' '1.5.10.129' *GEN)
    (1FFFFFFFFFFFF '1.5.10.20'
     '255.255.255.0' '1.5.10.129')))  
  TEXT('CAT4SVR Service processor')

This command creates a remote system configuration which uses the dynamic delivery method to configure the remote system. The system is identified using the CAT4SP service processor configuration specified by the SPNWSCFG parameter. The iSCSI remote interface for SCSI is configured using an IP address of 1.5.10.10 and a LAN address of 1.5.10.20. The iSCSI qualified name is automatically generated.

Error messages

*ESCAPE Messages

CPF2182
  Not authorized to library &1.

CPF90A8
  *SECADM special authority required to do requested operation.

CPF96C9
  Network server configuration &1 not created.
CPF9870
  Object &2 type *5 already exists in library &3.

CPFA1B8
  *IOSYSCFG authority required to use &1.
Create Network Server Desc (CRTNWSD)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Create Network Server Description (CRTNWSD) command creates a description for a network server. The description includes server software parameters, network protocol descriptions and definition of attached communications equipment (for example, line descriptions).

This command should be run from the Configure Network Server (CFGNWS) menu, which includes all of the steps needed to create an initial server configuration.

More information about using this command is in the Communications Configuration book, SC41-5401

More information about using this command when Server connection specified *IXSVR and Server operating system specified *WIN32 for the Network server type (TYPE) parameter can be found in the Windows environment on iSeries information in iSeries Information Center at http://www.iseries.ibm.com/infocenter.

Restrictions:
• You must have input/output system configuration (*IOSYSCFG) special authority to use this command.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>NWSD</td>
<td>Network server description</td>
<td>Communications name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td>RSRCNAME</td>
<td>Resource name</td>
<td>Name, *NONE, *AUTO</td>
<td>Required, Positional 2</td>
</tr>
<tr>
<td>TYPE</td>
<td>Network server type</td>
<td>Element list</td>
<td>Optional, Positional 3</td>
</tr>
<tr>
<td>STGPTH</td>
<td>Storage path</td>
<td>Values (up to 4 repetitions): Element list</td>
<td>Optional</td>
</tr>
</tbody>
</table>

© Copyright IBM Corp. 1998, 2006
<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
</table>
| MLTPTHGRP   | Multi-path group | Single values: *NONE  
Other values (up to 4 repetitions): 1-4 | Optional |
| DFTSTGPTH   | Default storage path | 1-4, 1, *MLTPTHGRP  
Optional |
| RMVMEDEPTH  | Removable media path | 1-4, 1, *MLTPTHGRP  
Optional |
| ACTTMR      | Activation timer | 30-1800, 120  
Optional |
| ONLINE      | Online at IPL | *YES, *NO  
Optional |
| VRYWAIT     | Vary on wait | 1-15, *NOWAIT  
Optional |
| SHUTDTIMO   | Shutdown timeout | 2-45, 15  
Optional |
| PARTITION   | Partition | Character value, *NONE  
Optional |
| PTNBR       | Partition number | Integer, *NONE  
Optional |
| DMNROLE     | Domain role | *DMNCTL, *BKUCTL, *SERVER  
Optional |
| PRPDMNUSR   | Propagate domain users | *YES, *NO  
Optional |
| LNGVER      | Language version | Integer, *PRIMARY  
Optional |
| CODEPAGE    | Code page | Integer, *LNGVER  
Optional |
| MSGQ        | Server message queue | Single values: *JOBLOG, *NONE  
Other values: Qualified object name | Optional |
| CMNMSGQ     | Communications message queue | Single values: *SYSPR, *NONE  
Other values: Qualified object name | Optional |
| CFGFILE     | Configuration file | Single values: *NONE  
Other values: Qualified object name | Optional |
| SVRSTGSIZE  | Server storage space sizes | Element list  
Optional |
| SVRSTGASP   | Server storage ASP | Element list  
Optional |
| STGASPDEV   | Server storage ASP device | Element list  
Optional |
<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
</table>
| TCPPORTCFG         | TCP/IP port configuration                 | Single values: *NONE  
Other values (up to 9 repetitions): Element list                          | Optional|
|                    | Element 1: Port                          | 1, 2, 3, 4, *INTERNAL, *VRTETHPTP, *VRTETH0,  
*VRTETH1, *VRTETH2, *VRTETH3, *VRTETH4,  
*VRTETH5, *VRTETH6, *VRTETH7, *VRTETH8,  
*VRTETH9                                                    |         |
|                    | Element 2: Internet address               | Character value                                                         |         |
|                    | Element 3: Subnet mask                    | Character value                                                         |         |
|                    | Element 4: Maximum transmission unit      | Integer, 1500                                                           |         |
|                    | Element 5: Gateway address                | Character value                                                         |         |
| TCPRTE             | TCP/IP route configuration                | Single values: *NONE  
Other values (up to 24 repetitions): Element list                          | Optional|
|                    | Element 1: Route destination              | Character value, *DFTROUTE                                               |         |
|                    | Element 2: Subnet mask                    | Character value, *NONE, *HOST                                            |         |
|                    | Element 3: Next hop                       | Character value                                                         |         |
| TCPHOSTNAM         | TCP/IP local host name                    | Character value, *NWSD                                                   | Optional|
| TCPDMNNAME         | TCP/IP local domain name                  | Character value, *SYS                                                    | Optional|
| TCPNAMSVR          | TCP/IP name server system                 | Single values: *SYS, *NONE  
Other values (up to 3 repetitions): Character value                           | Optional|
| PORTS              | Ports                                     | Single values: *NONE  
Other values (up to 5 repetitions): Element list                          | Optional|
|                    | Element 1: Port number                    | 1-2, *INTERNAL, *VRTETHPTP, *VRTETH0, *VRTETH1,  
*VRTETH2, *VRTETH3, *VRTETH4, *VRTETH5,  
*VRTETH6, *VRTETH7, *VRTETH8, *VRTETH9                       |         |
|                    | Element 2: Line description               | Name                                                                    |         |
| VRTETHPTH          | Virtual Ethernet path                     | Values (up to 5 repetitions): Element list                              | Optional|
|                    | Element 1: Port number                    | *VRTETHPTP, *VRTETH0, *VRTETH1, *VRTETH2,  
*VRTETH3, *VRTETH4, *VRTETH5, *VRTETH6,  
*VRTETH7, *VRTETH8, *VRTETH9                                          |         |
|                    | Element 2: Network server host adapter    | Communications name                                                     |         |
|                    | Element 3: IP security rules              | Element list                                                            |         |
|                    | Element 1: Remote interface 1 rule        | 1-16, *DFTSECRULE, *NONE                                               |         |
|                    | Element 2: Remote interface 2 rule        | 1-16, *DFTSECRULE, *NONE                                               |         |
|                    | Element 3: Remote interface 3 rule        | 1-16, *DFTSECRULE, *NONE                                               |         |
|                    | Element 4: Remote interface 4 rule        | 1-16, *DFTSECRULE, *NONE                                               |         |
| RSTDDEVRSC         | Restricted device resources               | Single values: *NONE, *ALL  
Other values (up to 10 repetitions): Name, *ALLTAPE,  
*ALLOPT                                                        | Optional|
<p>| NWSCFG             | Network server configuration              | Element list                                                            | Optional|
|                    | Element 1: Remote system name             | Name, *DFT                                                             |         |
|                    | Element 2: Connection security name       | Name, *DFT                                                             |         |
| SHUTDPORT          | Shutdown TCP port                         | 1024-65535, 8700                                                       | Optional|
| VRTETHCTLTP        | Virtual Ethernet control port             | 1024-65535, 8800                                                       | Optional|
| SYNCTIME           | Synchronize date and time                 | *TYPE, *YES, *NO                                                       | Optional|</p>
<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>IPLSRC</td>
<td>IPL source</td>
<td>*NWSSTG, *PANEL, *STMF, A, B, D</td>
<td>Optional</td>
</tr>
<tr>
<td>IPLSTMF</td>
<td>IPL stream file</td>
<td>Path name, *NONE</td>
<td>Optional</td>
</tr>
<tr>
<td>IPLPARM</td>
<td>IPL parameters</td>
<td>Character value, *NONE</td>
<td>Optional</td>
</tr>
<tr>
<td>PWRCTL</td>
<td>Power control</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
<tr>
<td>TEXT</td>
<td>Text ‘description’</td>
<td>Character value, *BLANK</td>
<td>Optional</td>
</tr>
</tbody>
</table>

**Network server description (NWSD)**

Specifies the network server description to be created.

This is a required parameter.

*communications-name*

Specify the name of the network server description. The name must be a valid communications name. The name cannot end with the character at code point X’5B’. That character is converted to a dollar sign ($) character in the ASCII character set, and network servers cannot have names ending in a dollar sign ($).

**Resource name (RSRCNAME)**

Specifies the resource name that identifies the hardware that the description represents. This is a required parameter.

**Note:** Use the Work with Hardware Resources (WRKHDWRSC) command with *CMN specified for the TYPE parameter to help determine the resource name. Specify the resource name of the input/output processor (IOP) or the input/output adapter (IOA) for the file server.

*NONE

A hardware resource is not associated with the network server. This value is only valid when **Server connection** specified *GUEST or *ISCSI for the Network server type (TYPE) parameter.

*ISCSI is the only valid value when *ISCSI is specified for **Server connection** on the Network server type (TYPE) parameter.

*AUTO

A hardware resource will be determined based on the partition. This value is only valid when **Server connection** specified *GUEST for the Network server type (TYPE) parameter.

**name** Specify a resource name.

**Network server type (TYPE)**

Specifies the type of network server description to create. This information consists of two parts including the Server connection and the Server operating system.

**Element 1: Server connection**
*IXSVR
Create an integrated network server description. The Server operating system value is required for *IXSVR.

*ISCSI
Create a network server description that uses an iSCSI connection. The Server operating system value is required for *ISCSI.

*GUEST
Create a network server description for a guest operating system running in a logical partition. The Server operating system value is optional for *GUEST.

Note: Specifying *GUEST is equivalent to specifying *GUEST with *LINUXPPC for the server operating system.

*WINDOWSNT
Create a Windows network server description. The Server operating system value is ignored for *WINDOWSNT.

Note: Specifying *WINDOWSNT is equivalent to specifying *IXSVR with *WIN32 for the server operating system.

Element 2: Server operating system

Note: This element is ignored when *WINDOWSNT is specified on the first element.

*WIN32
Create a network server description for a 32 bit Windows operating system.

*LINUX32
Create a network server description for a 32 bit Linux operating system.

*LINUXPPC
Create a network server description for a Linux Power PC operating system.

*AIXPPC
Create a network server description for an AIX Power PC operating system.

Storage path (STGPTH)

Specifies the storage paths the storage spaces can use. This information consists of two parts including the Network server host adapter description and the IP security rules for this path. You can enter up to four values for this parameter. You must enter at least one storage path.

Note: This parameter is only valid when Server connection specified *ISCSI for the Network server type (TYPE) parameter.

Element 1: Network server host adapter

name
Specify a name of network server host adapter (NWSH) description. The network server host adapter name must be unique for each STGPTH parameter on this NWSD.

*NONE
This storage path is not used by this network server.

Element 2: IP security rules

Specify any relative entry of the IP security rule (IPSECRULE) parameter, defined in NWS Configuration(NWSCFG) of type *CNNSEC, that will be used for each remote port’s security.
Element 1: Remote interface 1 rule

**DFTSECRULE**
Remote interface 1 will use IP security rule specified on the Default IP security rule (DFTSECRULE) parameter.

**NONE**
Remote interface 1 will not use IP security rule.

**1-16**
Remote interface 1 will use IP security rule specified.

Element 2: Remote interface 2 rule

**DFTSECRULE**
Remote interface 2 will use IP security rule specified on the DFTSECRULE parameter.

**NONE**
Remote interface 2 will not use IP security rule.

**1-16**
Remote interface 2 will use IP security rule specified.

Element 3: Remote interface 3 rule

**DFTSECRULE**
Remote interface 3 will use IP security rule specified on the DFTSECRULE parameter.

**NONE**
Remote interface 3 will not use IP security rule.

**1-16**
Remote interface 3 will use IP security rule specified.

Element 4: Remote interface 4 rule

**DFTSECRULE**
Remote interface 4 will use IP security rule specified on the DFTSECRULE parameter.

**NONE**
Remote interface 4 will not use IP security rule.

**1-16**
Remote interface 4 will use IP security rule specified.

---

**Default IP security rule (DFTSECRULE)**

Specify any defined entry of the IP security rules (IPSECRULE) parameter, defined in Network server configuration (NWSCFG) of type connection security (*CNNSEC), that will be used for storage and virtual Ethernet connections that are configured to use the default security rule.

**Note:** This parameter is only valid when Server connection specified *ISCSI for the Network server type (TYPE) parameter.

**NONE**
Remote interfaces will not use IP security rules.

**1-16**
Remote interfaces will use IP security rule specified.
Multi-path group (MLTPTHGRP)

Specify storage paths as defined in the Storage path (STGPTH) parameter.
1. See the Storage path (STGPTH) parameter to determine what storage paths are valid.
2. This parameter is only valid when Server connection specified *ISCSI for the Network server type (TYPE) parameter.

Single values

*NONE
   No multi-path group defined.

Other values (up to 4 repetitions)

1-4 Specify at least two relative storage paths.

Default storage path (DFTSTGPTH)

Specify a storage path as defined in the Storage path (STGPTH) parameter or specify the multi-path group as defined in the Multi-path group (MLTPTHGRP) parameter.
1. See the Storage path (STGPTH) parameter to determine what storage paths are valid.
2. This parameter is only valid when Server connection specified *ISCSI for the Network server type (TYPE) parameter.

1 Default storage path will be storage path 1.
2-4 Default storage path will be storage path 2, 3 or 4.

*MLTPTHGRP
   Default storage path will be the multi-path group storage path.

Removable media path (RMVMEDPTH)

Specify a storage path as defined in the Storage path (STGPTH) parameter or specify the multi-path group as defined in the Multi-path group (MLTPTHGRP) parameter.
1. See the Storage path (STGPTH) parameter to determine what storage paths are valid.
2. This parameter is only valid when Server connection specified *ISCSI for the Network server type (TYPE) parameter.

1 Removable media will use storage path 1.
2-4 Removable media will use storage path 2, 3 or 4.

*MLTPTHGRP
   Removable media will use the multi-path group storage path.

Activation timer (ACTTMR)

Specifies the amount of time (in seconds) the system will wait for the connection to be established to the remote server’s service processor and to power on the remote server.
Note: This parameter is only valid when Server connection specified *ISCSI for the Network server type (TYPE) parameter.

120 The activate time of 120 seconds is used.

**activation-timer**

Specify, in seconds, a value ranging from 30 through 1800.

---

**Online at IPL (ONLINE)**

Specifies whether this object is automatically varied on at initial program load (IPL).

**Notes:**

1. This parameter is ignored when Server connection specified *IXSVR or *ISCSI for the Network server type (TYPE) parameter. To have the Windows server vary on, use the Change TCP/IP Interface (CHGTCPIPFC) command and set the AUTOSTART parameter to *YES for the private LAN line description for the server or a startup program could be called to vary on the servers after IPL processing has completed.
2. When Server connection specified *GUEST for the Network server type (TYPE) parameter, it is recommended that *NO is specified. A startup program should then be called to vary on the servers after IPL processing has completed.
3. If more than one network server description is created for a file server resource, only one network server description should specify ONLINE(*YES). If more than one network server description specifies ONLINE(*YES), only the first description, in alphabetical order, is varied on during the IPL.

| **YES** | The network server is automatically varied on at IPL. All configuration objects attached to the network server will also be varied on. |
| **NO**  | This network server is not automatically varied on at IPL. |

---

**Vary on wait (VRYWAIT)**

Specifies whether the network server is varied on asynchronously or synchronously. For synchronous vary on, specifies how long the system waits for the vary on to complete.

**Note:** Vary on of a network server resets the IOP. The vary on wait time specifies time in addition to the reset time.

| **NOWAIT** | The system does not wait for the vary on to complete. The network server is varied on asynchronously. |
| **1-15**   | Specify the number of minutes to wait. The system waits until the network server is varied on, or until the specified time passes, before completing the Vary Configuration (VRYCFG) command. |

---

**Shutdown timeout (SHUTDTIMO)**

Specifies the network server shutdown timeout value in minutes. This is used to limit the amount of time that the network servers operating system is allowed to shutdown before the network server is varied offline.
A shutdown timeout value of 15 minutes is used.

Specify the number of minutes to wait. The system waits until the network servers operating system has shutdown successfully, or until the specified time passes before varying the network server offline.

**Partition (PARTITION)**

Specifies the name of the logical partition to be used by this network server. Up to 48 characters can be specified.

*Note:* When Server connection specified *GUEST for the Network server type (TYPE) parameter either the Partition (PARTITION) or Partition number (PTNNBR) parameter can be specified but both parameters cannot be specified. When Server connection specified *GUEST for the Network server type (TYPE) parameter and RSRCNAME is specified, the Partition (PARTITION) and Partition number (PTNNBR) parameter can be set to *NONE.

*NONE  
A partition name is not used by this network server.

**name**  
Specify the name of the partition to be used by this network server. The partition name PRIMARY cannot be specified.

**Partition number (PTNNBR)**

Specifies the number of the logical partition to be used by this network server.

**Notes:**

1. When Server connection specified *GUEST for the Network server type (TYPE) parameter either the Partition (PARTITION) or Partition number (PTNNBR) parameter can be specified but both parameters cannot be specified.
2. When Server connection specified *GUEST for the Network server type (TYPE) parameter and RSRCNAME is specified, the Partition (PARTITION) and Partition number (PTNNBR) parameter can be set to *NONE.

*NONE  
A partition number is not used by this network server.

**integer**  
Specify the number of the partition to be used by this network server.

**Domain role (DMNROLE)**

Specifies the domain controller role performed by this network server.

*Note:* This parameter is not valid when Server connection specified *GUEST or Server operating system specified *LINUX32 for Network server type (TYPE) parameter.

*DMNCTL  
This network server is a domain controller within its domain.
*BKUCtrl
This network server is a backup controller within its domain.

*SERVER
This network server is a stand alone server.

Propagate domain users (PRPDMNUSR)
Specifies whether domain user enrollment should be allowed or not for this network server description.

Note: This parameter is ignored when Server connection specified *GUEST or Server operating system specified *LINUX32 for Network server type (TYPE) parameter.

*YES  Propagation of domain user enrollment is allowed.

*NO   Propagation of domain user enrollment is not allowed.

Language version (LNGVER)
Specifies the language version of the network server product. To change the language version, a new network server description must be created specifying the desired language.

Note: This parameter is not valid when Server connection specified *GUEST or Server operating system specified *LINUX32 for Network server type (TYPE) parameter.

*PRIMARY  The language version for the currently installed primary national language is used.

integer
Specify the language version of the network server product to be used. The language must be one of the installed versions of the network server product. Use the Work with Licensed Programs (LICPGM) menu to determine the installed languages. Language versions are entered in the command as an integer value.

Supported language version values include:

2902  Estonia
2903  Lithuanian
2904  Latvian
2905  Vietnamese
2906  Laotian
2909  English Belgium
2911  Slovenian
2912  Croatian
2914  Serbian
2922  Portuguese
2923  Dutch Netherlands
2924  English
2925 Finnish
2926 Danish
2928 French
2929 German
2930 Japanese
2931 Spanish
2932 Italian
2933 Norwegian
2937 Swedish
2938 English Uppercase DBCS
2939 German Multinational Character Set
2940 French Multinational Character Set
2942 Italian Multinational Character Set
2950 English Uppercase
2954 Arabic
2956 Turkish
2957 Greek
2958 Icelandic
2961 Hebrew
2962 Japanese Kanji
2963 Belgian Dutch Multinational Character Set
2966 Belgian French Multinational Character Set
2972 Thai
2974 Bulgarian
2975 Czech
2976 Hungarian
2978 Polish
2979 Russian
2980 Brazilian Portuguese
2981 Canadian French Multinational Character Set
2984 English DBCS
2986 Korean
2987 Traditional Chinese
2989 Simplified Chinese
2992 Romanian
2994 Slovakian
2995 Albanian
Code page (CODEPAGE)

Specifies the ASCII code page representing the character set to be used by this network server. Only certain code pages can be used for a given country or region code.

*LNGVER

Specifies to use the default code page corresponding to the language version (LNGVER) selected. When Server connection specified *GUEST or Server operating system specified *LINUX32 for Network server type (TYPE) parameter, value 437 will be used.

integer

Specify the ASCII code page which represents the character set used by the network server. The code page values that can be used with each country or region code are:

437 United States
850 Multilingual
852 Latin 2 (Czechoslovakia, Hungary, Poland, countries of the former Yugoslavia)
857 Turkish
860 Portuguese
861 Iceland
862 Hebrew-speaking
863 Canada (French-speaking)
864 Arabic-speaking
865 Nordic
866 Russian
932 Japanese
934 Korean
938 Chinese
942 Japanese SAA
944 Korean SAA
948 Chinese SAA
950 Traditional Chinese (DBCS)
1381 Simplified Chinese (DBCS)
Server message queue (MSGQ)

Specifies the message queue to receive server messages.

For details on the type of messages that are sent to this message queue, see the appropriate manual that is associated with the type of network server.

**Note:** When a value other than *NONE is specified, all server activity will be logged to either the monitor job log or the specified message queue. You should take the appropriate steps to secure this information on the iSeries.

One method of restricting access to the server information on the iSeries is to create a message queue to contain the server activity. This message queue should be created with AUT(*EXCLUDE) and then any users that are to have access to the server activity can be granted explicit authority to the message queue using the GRTOBJAUT command. Specify this message queue for this parameter.

Single values

*JOBLOG  Causes messages from the server to be placed on the joblog of the monitor job.

*NONE  Causes messages to not be placed on any message queue.

Qualifier 1: Server message queue

*name Specify the name of a message queue to receive messages issued by the server.

Qualifier 2: Library

*LIBL  All libraries in the library list for the current thread are searched until the first match is found.

*CURLIB  The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

*name Specify the name of the library to be searched.

Event log (EVTLOG)

Specifies whether or not messages from the event logs are received from the server.

Notes:

1. Event log messages are placed in the message queue that is identified by the Server message queue (MSGQ) parameter. The MSGQ value cannot be *NONE if a value other than *NONE is specified for this parameter. See the MSGQ parameter description for more information.

2. This parameter is not valid when Server connection specified *GUEST or Server operating system specified *LINUX32 for Network server type (TYPE) parameter.

Single values

*ALL  All the event log messages are received.

*NONE  No event log messages are received.

Other values (up to 3 repetitions)
*SYS  The system event log messages are received.
*SEC  The security event log messages are received.
*APP  The application event log messages are received.

**Communications message queue (CMNMSGQ)**

Specifies the name of a message queue to receive communications status messages. This parameter is only valid for server connection type *ISCSI.

For details on the type of messages that are sent to this message queue, see the appropriate manual that is associated with the type of network server.

**Single values**

*SYSOPR  Causes messages from the server to be placed in the system operator message queue.

**Qualifier 1: Communications message queue**

*name  Specify the name of a message queue to receive status messages issued by the server.

**Qualifier 2: Library**

*LIBL  All libraries in the library list for the current thread are searched until the first match is found.

*CURLIB  The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

*name  Specify the name of the library to be searched.

**Configuration file (CFGFILE)**

Specifies the source file containing configuration data to be used in activating or further defining the server.

**Note:** This parameter is not valid when **Server connection** specified *GUEST for **Network server type** (TYPE) parameter.

**Single values**

*NONE  No configuration file is specified.

**Qualifier 1: Configuration file**

*name  Specify the name of the source file containing the configuration data members for the server. At the time the server is activated, all members in the file are processed. The file must exist on the system by the time the server is activated.

**Qualifier 2: Library**

*LIBL  All libraries in the library list for the current thread are searched until the first match is found.
*CURLIB

The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

name Specify the name of the library to be searched.

---

**Server storage space sizes (SVRSTGSIZE)**

Specifies the size of the storage spaces, in megabytes.

**Note:** This parameter is not valid when Server connection specified *GUEST for Network server type (TYPE) parameter.

**Element 1: Install source size**

Specifies the size of the drive that holds the files that are used to install the server.

*DFT The default drive size is to be used.

integer Specifies the number of megabytes to use for the install source drive size. Valid values range from 200 to 2047 megabytes.

**Element 2: System size**

Specifies the size of the drive that the Windows server is installed on.

*DFT The default drive size is to be used.

integer Specifies the number of megabytes to use for the system drive size. Valid values range from 500 to 1024000 megabytes.

---

**Server storage space ASP (SVRSTGASP)**

Specifies the auxiliary storage pool (ASP) identifiers for the storage space that will contain the files used to install the Windows server and the storage space that will contain the Windows server operating system.

**Note:** This parameter is not valid when Server connection specified *GUEST for Network server type (TYPE) parameter.

**Note:** You cannot specify both a SVRSTGASP and STGASPDEV parameter value for the same element.

**Element 1: Install source ASP**

Specifies the auxiliary storage pool for the storage space object that holds the files that are used to install the Windows server.

1 The storage space is created in auxiliary storage pool 1, the system auxiliary storage pool.

2-255 Specify the number of the ASP to be used. Valid values depend on what ASPs are defined on the system.
Element 2: System ASP

Specifies the auxiliary storage pool for the storage space object that holds the Windows server operating system.

1 The storage space is created in auxiliary storage pool 1, the system auxiliary storage pool.

2-255 Specify the number of the ASP to be used. Valid values depend on what ASPs are defined on the system.

Server storage ASP device (STGASPDEV)

Specifies the auxiliary storage pool (ASP) device for the storage space that will contain the files used to install the Windows server and the storage space that will contain the Windows server operating system.

**Note:** This parameter is not valid when Server connection specified *GUEST for Network server type (TYPE) parameter.

**Note:** The ASP must have been activated (by varying on the ASP device) and have a status of ‘Available’.

**Note:** You cannot specify both a SVRSTGASP and STGASPDEV parameter value for the same element.

Element 1: Install source ASP device

Specifies the auxiliary storage pool device name for the storage space object that holds the files that are used to install the Windows server.

`name` Specify the device name of the ASP to use for the network server storage space.

Element 2: System ASP device

Specifies the auxiliary storage pool (ASP) device for the storage space object that holds the Windows server operating system.

`name` Specify the device name of the ASP to use for the network server storage space.

TCP/IP port configuration (TCPPORTCFG)

Specifies the TCP/IP configuration values that are specific to a port on the network server. This information consists of four parts including the identification of the network server port, the internet address assigned to the port and the subnet mask of the port. You can enter up to nine values for this parameter.

**Single values**

**NONE** There is no TCP/IP port configuration. *NONE cannot be specified when Server connection specified *IXSVR or *ISCSI for the Network server type (TYPE) parameter.

Element 1: Port

1 Network server port number 1 is configured.

2 Network server port number 2 is configured.
Network server port number 3 is configured.

Network server port number 4 is configured.

*INTERNAL
The network server internal token ring port is configured.

Note: *INTERNAL is not valid when Server connection specified *ISCSI or *GUEST for the Network server type (TYPE) parameter in the corresponding CRTNWSD command.

*VRTETHPTP
The network server virtual Ethernet point to point port is configured.

*VRTETHn
Virtual Ethernet port ‘n’ is configured, where ‘n’ is a number from 1 to 9.

Element 2: Internet address

Specify the local internet address which the network server responds to when Server connection specified *IXSVR or *ISCSI for the Network server type (TYPE) parameter. The internet address is specified in the form, nnn.nnn.nnn.nnn, where nnn is a decimal number ranging from 0 through 255. An internet address that has a binary value of all ones or all binary zeros for the network identifier (ID) portion or the host ID portion of the Internet address is not valid. The internet address selected must be unique across all network server descriptions and the i5/OS TCP/IP configuration.

Element 3: Subnet mask

Specify the subnet mask associated with the network server port. See the TCP/IP Fastpath Setup book for general information about subnets.

Subnetting provides the capability to partition an internet domain. Specify the mask for the network subnet and host address fields of the internet address that defines a subnet. The subnet mask is in the form, nnn.nnn.nnn.nnn, where nnn is a decimal number ranging from 0 through 255. The subnet mask must mask off all bits of the network class’s network ID portion of the internet address. For example, a subnet mask of 255.255.255.0 defines a Class B subnet consisting of all bits in the network portion of the internet address (this is a given) and consisting of all bits in the third byte of an internet address.

Element 4: Maximum transmission unit

The maximum transmission unit (MTU) value is 1500 bytes.

Specify the MTU value for the interface.

Element 5: Gateway address

Specify the default gateway address for the internet address in the form, nnn.nnn.nnn.nnn, where nnn is a decimal number ranging from 0 through 255.
TCP/IP route configuration (TCPRTE)

Specifies routes to remote destination systems or networks to the Transmission Control Protocol/Internet Protocol (TCP/IP) configuration for the network server. A route specification has three elements: the route destination, the subnet mask, and the next hop internet address. A maximum of 24 route specifications can be specified.

Note: This parameter will be ignored when Server connection specified *IXSVR or *ISCSI for the Network server type (TYPE) parameter.

Two values uniquely define a route. They are the route destination field and the subnet mask. For *DFTRoute values, the next hop element uniquely defines the route.

Single values

*NONE

There is no routing specification needed for the network server. *NONE must be specified when TCPPORTCFG(*NONE) is specified. *NONE may be specified if there is no need for route specifications.

Element 1: Route destination

Specifies the remote network or host that is being added. The user must specify all four bytes that make up an internet address though some of the bytes may be equal to 0. For example, a route to all the hosts on the 9.5.11 subnet is identified by entering 9.5.11.0 for the route destination. Used in combination with a subnet mask, the route destination will identify a route to a network or system.

*DFTRoute

A TCP/IP default route is being added. A default route entry is used by the system to route data that is being sent to an undefined network or system. Multiple *DFTRoute entries may be specified. The *DFTRoute entries are used in the order specified. If a particular next hop gateway on a *DFTRoute entry is not available, then the subsequent *DFTRoute entry’s next hop gateway specified will be used. This will continue until a *DFTRoute entry’s gateway is found that is active or the list of next hop gateway values is exhausted.

Character-value

Specify the route destination being added. The route destination can be specified in the form, mnn.0.0.0 for Class A, mnn.mnn.0.0 for Class B, and mnn.mnn.mnn.0 for Class C, or mnn.mnn.mnn.mnn for any combination thereof, where mnn is a decimal number ranging from 0 through 255.

Note: Any combination thereof means that you may specify a route, such as 9.5.0.0 to the hosts on the 9.5 subnet even though all 9.5.x.x addresses are class A network addresses.

Exceptions:

- The first byte (octet) must be greater than 0 and less than 255
- The last byte (octet) may not equal 255.
- The last byte (octet) may not equal 0 if *HOST is specified for the SUBNETMASK value.
- Routes to a broadcast address are not allowed.

Element 2: Subnet mask

A subnet mask value must be specified if *DFTRoute or a route destination is entered for the route destination element. Subnet mask specifies a bit mask that identifies to TCP/IP which bits of the value specified for the route destination compose the network and subnet portions of the internet address. The subnet is identified by combining the route destination internet address and the subnet mask.
*NONE

There is no subnet mask. If *DFTROUTE is specified in the route destination element, then *NONE must be specified. *NONE is valid only for the *DFTROUTE route destination value.

*HOST

The internet address value specified in the route destination field is a host address. The subnetmask value is calculated to be 255.255.255.255.

character-value

Specify the mask of the subnet field. The internet address is in the form, nnn.nnn.nnn.nnn, where nnn is a decimal number ranging from 0 through 255. For example, a destination route’s internet address value of 129.35.11.0 is a Class B subnet. The network ID part of its address is 129.35. The upper 2 bytes must designate 255 in the subnet mask, for example, the subnet mask must appear like 255.255.x.x, where x is determined by the user. The portion of the subnet mask which is associated with the network portion of a particular class of address must equal 255.

Element 3: Next hop

The next hop value specifies the internet address of the next system (gateway) on the route. A route cannot be added unless the internet address specified by the next hop element is directly reachable through a network associated with one of the network server ports.

See the Fastpath for TCP/IP book for general information about internet addresses.

character-value

Specify the internet address of the next system on the route in the form, nnn.nnn.nnn.nnn, where nnn is a decimal number ranging from 0 through 255 except that the host ID portion and the network ID portion of the internet address may not be all 0 bits or all 1 bits. An internet address that has all binary ones or all binary zeros for the network ID portion or the host ID portion of the internet address is not valid.

TCP/IP local host name (TCPHOSTNAM)

Specifies the short form of the host name to be associated with the network server.

The host name can be a text string having 2 through 63 characters.

The following characters are allowed in host names:

- Alphabetical characters A through Z
- Digits 0 through 9
- Minus sign (-)

*NWSD

Specifies that the host name for the network server is the same as the name of the network server description. *NWSD must be specified if *NONE is specified for the TCP/IP port configuration (TCPPORTCFG) parameter.

name Specifying a host name to be associated with the network server.

TCP/IP local domain name (TCPDMNNNAME)

Specifies the local domain name associated with the network server.
A domain name can be a text string having 2 to 255 characters. Domain names consist of one or more labels separated by periods. Each label can contain up to 63 characters. The following characters are allowed in domain names:

- Alphabetic characters A through Z
- Digits 0 through 9
- Minus sign (-)
- Period (.). Periods are only allowed when they separate labels of domain style name (refer to RFC 1034).

Other domain name conventions include the following:

- Uppercase and lowercase characters are allowed, but no significance attached to the case. The case is maintained as entered. The first and last character of the host name must be an alphabetic character or a digit.
- Try to limit your domain name labels to 12 characters because shorter labels are easier to remember.
- It is a common practice to use hierarchical names that allow predictable extensions for change and growth. Domain names normally reflect the delegation of authority or hierarchy used to assign them.

For example, the name SYS1.MFG.ABC.COM can be broken down into the following:

COM  All commercial networks.

ABC.COM  All systems in the ABC company’s commercial network.

MFG.ABC.COM  All manufacturing systems in the ABC company’s commercial network.

SYS1.MFG.ABC.COM  A host named SYS1 in the manufacturing area of the company’s commercial network.

In the above example, MFG.ABC.COM is the domain name and SYS1 is the short form of the host name.

The COM designation is one of several domain names used when connecting the Internet. Some of the other domain names are as follows:

COM  Commercial organizations
EDU  Educational institutions
GOV  Government institutions
MIL  Military groups
NET  Major network support centers
ORG  Organizations other than those listed above

Country code
- Countries or regions other than USA

*SYS  Specifies that the local domain name for the network server should be the same value as is configured for the iSeries system. *SYS must be specified if TCPPORTCFG(*NONE) is specified. Also, *SYS must be specified if only an *INTERNAL port is specified on the TCPPORTCFG parameter.

character-value
- Specify a host name to be associated with the network server.
TCP/IP name server system (TCPNAMSVR)

Specifies the internet address of the name server system that is used by the network server. Typically, this is the same value as it is for the iSeries.

Single values

*SYS  The name server system used by the network server should be the same as for the iSeries. *SYS must be specified if TCPPORTCFG(*NONE) is specified.

*NONE  There is no name server to be used by the network server.

Other values (up to 3 repetitions)

character-value  Specify an internet address for the name server system to be used by the network server. Up to three remote name server systems can be specified. The name server systems are used in the order they are specified.

Ports (PORTS)

Specifies the names of the lines attached to the *INTERNAL port, *VRTETH port or to the two line ports on the network server.

Note: This parameter is not valid when Server connection specified *GUEST for Network server type (TYPE) parameter.

*NONE  No lines are attached to this server. Lines may be attached later by specifying this server description in the line descriptions when they are created.

Element 1: Port number

*INTERNAL  If *INTERNAL is specified, then the line description must be the name of a token ring network (TRN). Also, *INTERNAL can only be specified for one token ring line description. This value is not valid when Server connection specified *ISCSI for the Network server type (TYPE) parameter in the corresponding CRTNWSD command.

*VRTETHPTP  The network server virtual Ethernet point to point port is configured.

*VRTETHn  Virtual Ethernet port ‘n’ is configured, where ‘n’ is a number from 1 to 9.

1  Network server port number 1 is configured. This value is not valid for server connection type *ISCSI.

2  Network server port number 2 is configured. This value is not valid for server connection type *ISCSI.

Element 2: Line description

name  Specify the name of an existing line description. The name of a token ring network (TRN) or an Ethernet (ETH) line description can be specified. The line must have been created specifying RSRCNAME(*NWSD), and must not be currently attached to another server. The name of a token ring network (TRN) line description is not valid for server connection type *ISCSI.
Virtual Ethernet path (VRTETHPTH)

Specifies the virtual Ethernet paths the Ethernet line descriptions can use. This information consists of three parts including the virtual Ethernet port, the Network server host adapter description and the IP security rules for this path. You can enter up to five values for this parameter. You must enter at least one virtual Ethernet path which is the path to be used by the *VRTETHPTH line description.

**Note:** This parameter is only valid when Server connection specified *ISCSI for the Network server type (TYPE) parameter.

Element 1: Port number

*VRTETHPTP

The network server virtual Ethernet point to point port is configured.

*VRTETHn
Virtual Ethernet port ‘n’ is configured, where ‘n’ is a number from 1 to 9.

Element 2: Network server host adapter

**name** Specify the name of an existing network server host adapter (NWSH) description. The network server host adapter name does not need to be unique for each VRTETHPTH parameter on this NWSD.

Element 3: IP security rules

Element 1: Remote interface 1 rule

*DFTSECRULE
Remote interface 1 will use IP security rule specified on the Default IP security rule (DFTSECRULE) parameter.

*NONE
Remote interface 1 will not use IP security rule.

1-16 Remote interface 1 will use IP security rule specified.

Element 2: Remote interface 2 rule

*DFTSECRULE
Remote interface 2 will use IP security rule specified on the DFTSECRULE parameter.

*NONE
Remote interface 2 will not use IP security rule.

1-16 Remote interface 2 will use IP security rule specified.

Element 3: Remote interface 3 rule

*DFTSECRULE
Remote interface 3 will use IP security rule specified on the DFTSECRULE parameter.

*NONE
Remote interface 3 will not use IP security rule.

1-16 Remote interface 3 will use IP security rule specified.

Element 4: Remote interface 4 rule
*DFTSECRULE
Remote interface 4 will use IP security rule specified on the DFTSECRULE parameter.

*NONE
Remote interface 4 will not use IP security rule.

1-16 Remote interface 4 will use IP security rule specified.

---

**Restricted device resources (RSTDDEVRSC)**

Specifies the iSeries tape and optical device resource names that are restricted and cannot be used by the network server.

The resource is used when the network server is active and a request is issued from a client application running on the network server. The device resource cannot be used by the application and the i5/OS at the same time. If the device resource is in use by the i5/OS, the network server application will not use this resource. If the device resource is intended to be used by the network server application, it will need to be available when the network server application is ready to use it.

**Notes:**
1. Only tape and optical device resources can be restricted.
2. If other device resources are specified that are not valid or are not detected, they will not allow the network server to vary on.

**Single values**

*NONE
No device resources are restricted from the network server. Therefore, any tape or optical device resources that exist on the system can be used.

*ALL All tape and optical device resources are restricted from being used by the network server.

**Other values (up to 10 repetitions)**

*ALLOPT
All optical device resources are restricted from being used by the network server.

**Note:** This value can only be specified once.

*ALLTAPE
All tape resources are restricted from being used by the network server.

**Note:** This value can only be specified once.

*name Specify the resource name of a restricted device that cannot be used by the network server. Up to 10 restricted device resource names can be specified.

---

**Network server configuration (NWSCFG)**

Specifies the network server configuration descriptions to use with this NWSD. This information consists of two parts including the Remote system name and Connection security name.
Note: This parameter is only valid when Server connection specified *ISCSI for the Network server type (TYPE) parameter.

Element 1: Remote system name

*DFT Use the system generated default remote system network server configuration name of 'nwsdnameRM' where nwsdname is the name of this network server description.

name Specify the name of an existing remote system network server configuration description.

Element 2: Connection security name

*DFT Use the system generated default connection security network server configuration name of 'nwsdnameCN' where nwsdname is the name of this network server description.

name Specify the name of an existing connection security network server configuration description.

---

## Shutdown TCP port (SHUTDPORT)

Specifies the TCP port to use for shutdown.

This port listens through the local area network (LAN) interface configured for any network server host adapter (NWSH) device associated with the network server description (NWSD) object on the Virtual Ethernet path (VRTETHPTH) parameter.

Note: This parameter is only valid when Server connection specified *ISCSI for the Network server type (TYPE) parameter.

8700 Use the TCP port number of 8700.

integer Specifies the port number identifying the port that is to be used for shutdown. Valid values range from 1024 through 65,535.

---

## Virtual Ethernet control port (VRTETHCTLP)

Specifies the TCP port to use for virtual Ethernet control.

This port listens through the local area network (LAN) interface configured for any network server host adapter (NWSH) device associated with the network server description (NWSD) object on the Virtual Ethernet path (VRTETHPTH) parameter.

Note: This parameter is only valid when Server connection specified *ISCSI for the Network server type (TYPE) parameter.

8800 Use the TCP port number of 8800.

integer Specifies the port number identifying the port that is to be used for virtual Ethernet control. Valid values range from 1024 through 65,535.
Synchronize date and time (SYNCTIME)

Specifies whether the iSeries should synchronize the network server date and time with the iSeries system date and time.

**Note:** This parameter is not valid when **Server operating system** specified *LINUX32 for Network server type (TYPE) parameter.

*TYPE* The iSeries will perform synchronization based on the network server type. If **Server operating system** specified *WIN32 for the Network server type (TYPE) parameter, synchronization will be done as if SYNCTIME(*YES) was specified.

*YES* The iSeries will synchronize the network server date and time with the iSeries system date and time.

The following occurs if **Server operating system** specified *WIN32 for the Network server type (TYPE) parameter:
- If the network server description is varied on, this reset will occur immediately and at least every 30 minutes thereafter.
- If the network server description is varied off, this reset will occur when the network server description is varied on and at least every 30 minutes thereafter.

The QTIMZON system value must be set to the correct value for time synchronization to work correctly.

*NO* The iSeries synchronizes the network server date and time with the iSeries system date and time when the network server description is varied on, but will not keep the date and time synchronized while the network server description is varied on. If the network server description is varied on when this parameter is set to this value, the date and time will no longer be synchronized while the network server description is varied on.

IPL source (IPLSRC)

Specifies the source of the load image that the partition is started from.

**Note:** This parameter is only valid when **Server connection** specified *GUEST for Network server type (TYPE) parameter.

*NWSSTG* The partition is started using the load image in the first network server storage space attached to this network server description.

*STMF* The partition is started using the load image in the stream file specified by the IPL stream file (IPLSTMF) parameter.

*PANEL* The partition is started from the source indicated on the operator’s panel.

A The partition is started from the A-source.
B The partition is started from the B-source.
D The partition is started from the D-source.
IPL stream file (IPLSTMF)

Specifies the path of the stream file containing the image that the partition should be loaded from.

**Note:** This parameter is only valid when Server connection specified *GUEST for Network server type (TYPE) parameter, and *STMF is specified for the IPL source (IPLSRC) parameter.

*NONE

A stream file is not specified.

**path-name**

Specify the path of the stream file containing the load image. Up to 5000 characters may be specified.

IPL parameters (IPLPARM)

Specifies a string of characters that will be passed to the load image at IPL time. It consists of commands or configuration information for the guest operating system.

**Note:** This parameter is only valid when Server connection specified *GUEST for Network server type (TYPE) parameter.

*NONE

IPL parameters are not passed to the load image.

**character-value**

Specify a string of up to 256 characters containing the IPL parameters to be passed to the load image.

Power control (PWRCTL)

Specifies whether the partition associated with the network server description will be powered down when the network server description is varied offline or powered up when the network server description is varied online.

**Note:** This parameter is only valid when Server connection specified *GUEST for the Network server type (TYPE) parameter.

*YES

The partition associated with the network server description will be powered down when the network server description is varied offline or powered up when the network server description is varied online.

*NO

The partition associated with the network server description will not be powered down when the network server description is varied offline and the partition will not be powered up when the network server description is varied online.

Authority (AUT)

Specifies the authority you are giving to users who do not have specific authority for the object, who are not on an authorization list, and whose group profile or supplemental group profiles do not have specific authority for the object.
*CHANGE

The user can perform all operations on the object except those limited to the owner or controlled by object existence (*OBJEXIST) and object management (*OBJMGT) authorities. The user can change and perform basic functions on the object. *CHANGE authority provides object operational (*OBJOPR) authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove users.

*ALL

The user can perform all operations except those limited to the owner or controlled by authorization list management (*AUTLMGT) authority. The user can control the object’s existence, specify the security for the object, change the object, and perform basic functions on the object. The user also can change ownership of the object.

*USE

The user can perform basic operations on the object, such as running a program or reading a file. The user cannot change the object. Use (*USE) authority provides object operational (*OBJOPR), read (*READ), and execute (*EXECUTE) authorities.

*EXCLUDE

The user cannot access the object.

*LIBCRTAUT

The system determines the authority for the object by using the value specified for the Create authority (CRTAUT) parameter on the Create Library (CRTLIB) command for the library containing the object to be created. If the value specified for the CRTAUT parameter is changed, the new value will not affect any existing objects.

name

Specify the name of an authorization list to be used for authority to the object. Users included in the authorization list are granted authority to the object as specified in the list. The authorization list must exist when the object is created.

Text 'description' (TEXT)

Specifies the text that briefly describes the network server description.

*BLANK

Text is not specified.

character-value

Specify no more than 50 characters of text, enclosed in apostrophes.

Examples

Example 1: Creating a *GUEST NWSD

CRTNWSD NWSD(LINUX1) RSRCNAME(*NONE) TYPE(*GUEST) PARTITION(TEST) VRYWAIT(*NOWAIT) CODEPAGE(437) TCPPORTCFG((1 '9.5.3.2' '255.255.255.0' 2048)) TCPPORTNAME(*NWSD) TCPDMNNNAME(*SYS) TCPNAMSVR(*SYS)

This command creates a server description named LINUX1. LINUX1 has no associated resource name. LINUX1 is a network server description associated with a guest operating system running in the logical partition named TEST. The TCP/IP protocol stack will be activated when LINUX1 is varied on. Code page 437 (United States) will be used. Port 1 will have TCP/IP internet addresses assigned. The TCP/IP local host name is the same as the server description name. The TCP/IP local domain name is the same as the i5/OS system and the same name server will be used.
Example 2: Creating a Windows *ISCSI NWSD

CRTNWSD NWSD(WINISCSI) RSRCNAME(*NONE) TYPE(*ISCSI *WIN32)
   STGPTH((NWSH1) (NWSH2 (1 1 1))) DFTSECRULE(3)
   MLTPTHGRP(1 2) DFTSTGPTH(1) RMVEDPTH(2) ACTMR(120)
   CMNMSGQ(MYLIB/MYMSMQ)
   TCPPORTCFG(*VRTETHPTP '9.5.5.5' '255.255.255.0')
   VRTETHPTH(*VRTETHPTP NWSH1) (*VRTETH5 NWSH2)
   NWSCFG(MYREMOTE MYSECURITY)
   SHUTDPORT(8750) VRTETHCTL(8850)

This command creates a server description named WINISCSI. WINISCSI has no resource name. WINISCSI is a network server description which has NWSH1 and NWSH2 network server host adapters and associated IP security rules as its Storage paths, Default IP security rule 3, Multi-path group 1 and 2, Default storage path 1, Removable media path 2, Activation timer 120 seconds, Communications message queue MYMSGQ in MYLIB, Virtual Ethernet paths NWSH1 and NWSH2, Network server configuration MYREMOTE for Remote system name and MYSECURITY for Connection security name, Shutdown TCP port 8750, and Virtual Ethernet control TCP port 8850.

Error messages

*ESCAPE Messages

CPF26AC
   Network server description &1 not created due to errors.
Create NWS Storage Space (CRTNWSSTG)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Create Network Server Storage Space (CRTNWSSTG) command creates a storage space used by a network server. The network storage space must be linked to a network server description before it can be used. For more information see the Add Network Server Storage Link (ADNWSSTGL) command.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>NWSSTG</td>
<td>Network server storage space</td>
<td>Name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td>NWSSIZE</td>
<td>Size</td>
<td>Integer, *CALC</td>
<td>Optional, Positional 2</td>
</tr>
<tr>
<td>FROMNWSSTG</td>
<td>From storage space</td>
<td>Name, *NONE</td>
<td>Optional</td>
</tr>
<tr>
<td>ASP</td>
<td>Auxiliary storage pool ID</td>
<td>1-255, 1</td>
<td>Optional</td>
</tr>
<tr>
<td>ASPDEV</td>
<td>ASP device</td>
<td>Name</td>
<td>Optional</td>
</tr>
<tr>
<td>CLUDMN</td>
<td>Cluster domain name</td>
<td>Character value</td>
<td>Optional</td>
</tr>
<tr>
<td>CLUPORTCFG</td>
<td>Cluster port configuration</td>
<td>Element list</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Element 1: Connection port</td>
<td>*VRTETH0, *VRTETH1, *VRTETH2, *VRTETH3,</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>*VRTETH4, *VRTETH5, *VRTETH6, *VRTETH7,</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>*VRTETH8, *VRTETH9</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Cluster internet address</td>
<td>Character value</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 3: Cluster subnet mask</td>
<td>Character value</td>
<td></td>
</tr>
<tr>
<td>TEXT</td>
<td>Text ‘description’</td>
<td>Character value, *BLANK</td>
<td>Optional</td>
</tr>
</tbody>
</table>

Network server storage space (NWSSTG)

Specify the name of the network server storage space to be created.

Size (NWSSIZE)

Specifies the size of the network server storage space to be created.
*CALC

If the FROMNWSSTG parameter is *NONE, and a format of *NTFS is specified, the storage space size will be set to '2' MB. When *FAT32 is specified, the storage space size will be set to '512' MB. When *FAT or *OPEN is specified, the size will be set to '1' MB. When *NTFSQR is specified, the storage space size will be set to '500' MB.

If a valid network server storage space name has been specified in the FROMNWSSTG parameter, the new storage space will be created with the same size as the storage space specified in the FROMNWSSTG parameter.

1-1024000

Specify the size for the network server storage space, in megabytes.

- The range for FORMAT types *NTFS, *FAT, *FAT32, and *OPEN is from 1 to 1024000 megabytes.
- The range for FORMAT(*NTFSQR) is from 500 to 1024000 megabytes.

When a FROMNWSSTG network server storage space is specified, the NWSSIZE parameter must be set equal to or greater than the size of the network server storage space specified in the FROMNWSSTG parameter.

---

From storage space (FROMNWSSTG)

Specifies the name of an existing network server storage space that will be copied to the new network server storage space.

*NONE

A copy operation from an existing network server storage space will not be performed.

'name'

An existing network server storage space name used to copy to the new network server storage space being created.

---

Format (FORMAT)

When a storage space is initially created, it is not formatted by the system. The storage space will need to be linked to a network server description and formatted by the hosted operating system.

*NTFS

The storage space should be formatted using Windows NT File System. Storage spaces created with this format can be linked to network server descriptions with connection types of *IXSVR, *ISCSI or *GUEST. This option offers the greatest benefit in a Windows server environment because of its better performance and integrated support of long filenames, larger disks, extended file attributes, file-security and recoverability features.

This option offers the greatest benefit in a Windows server environment because of its better performance and integrated support of long filenames, larger disks, extended file attributes, file-security and recoverability features. The size (NWSSIZE) parameter for a NTFS storage space must be at least 2 megabytes.

*FAT

The storage space should be formatted using the File Allocation Table file system. Storage spaces created with this format can be linked to network server descriptions with connection types of *IXSVR, *ISCSI, or *GUEST.

In most cases, however, greater efficiency and space utilization are achieved using the NTFS file system for Windows operating system types.
*FAT32
The storage space should be formatted using the 32-bit File Allocation Table file system. Storage spaces created with this format can be linked to network server descriptions with connection types of *IXSVR, *ISCSI, or *GUEST.

It includes support of long filenames and larger disk sizes. The size (NWSSIZE) parameter for a FAT32 storage space must be at least 512 megabytes.

*OPEN
The storage space should be formatted with an open file source file system. Storage spaces with this format can be linked to network server descriptions with Linux or AIX operating systems only.

*NTFSQR
The storage space should be formatted using Windows NT File System. Storage spaces created with this format will contain special attributes which makes it only linkable as a quorum resource disk used for Windows clustering.

---

**Auxiliary storage pool ID (ASP)**

Specifies the auxiliary storage pool (ASP) that will contain the new network server storage space.

**Note:** You cannot specify a value for both the ASP and ASPDEV parameters.

<table>
<thead>
<tr>
<th>'1'</th>
<th>The network server storage space is created in the system auxiliary pool ASP 1.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ASP-number</strong></td>
<td>The network server storage space is created in user auxiliary storage pool 2-32 or in independent auxiliary storage pools 33-255.</td>
</tr>
</tbody>
</table>

---

**ASP device (ASPDEV)**

Specifies the name of the auxiliary storage pool (ASP) device where storage is allocated for the network server storage space.

**Note:** The ASP must have been activated (by varying on the ASP device) and have a status of 'Available'.

**Note:** You cannot specify a value for both the ASP and ASPDEV parameters.

| **ASP-device-name** | The device name of the ASP to use for the network server storage space. |

---

**Cluster domain name (CLUDMN)**

Specifies the domain name of the cluster. This is the domain where the cluster service account will be created.

**Note:** This parameter is required when FORMAT(*NTFSQR) is specified.
Cluster port configuration (CLUPORTCFG)

Specifies the TCP/IP configuration values that are specific to the cluster service. This information consists of three parts including the identification of the cluster connection port, the cluster internet address, and the cluster subnet mask.

Note: This parameter is required when FORMAT(*NTFSQR) is specified.

Element 1: Connection port

Specifies the virtual ethernet port to be configured for this cluster. This connection will be used as a private cluster connection between each cluster node.

Element 2: Cluster internet address

Specifies the internet address of this cluster. The internet address is specified in the form, nnn.nnn.nnn.nnn where nnn is a decimal number ranging from 0 through 255. An internet address that has a binary value of all ones or all zeros for the network identifier (ID) portion or the host ID portion of the internet address is not valid.

Element 3: Cluster subnet mask

Specifies the subnet mask associated with cluster internet address. Subnetting provides the capability to partition an internet domain. Specify the mask for the network subnet and host address fields of the internet address that defines a subnet. The subnet mask is in the form, nnn.nnn.nnn.nnn, where nnn is a decimal number ranging from 0 through 255.

Text ’description’ (TEXT)

Specifies text describing the storage space.

*BLANK

Text is not specified.

’description’

Specify no more than 50 characters of text, enclosed in apostrophes.

Examples

Example 1: Create NTFS-format Storage Space

CRTNWSSTG NWSSTG(STGSPACE3) NWSSIZE(200) FORMAT(*NTFS)

This command creates a network server storage space called STGSPACE3 with a size of 200 megabytes. The storage space will not be formatted and must be linked to a network server description with a connection type of *IXSVR, *ISCSI or *GUEST.

Example 2: Copy Existing Storage Space

CRTNWSSTG NWSSTG(STGSPACE4) NWSSIZE(*CALC) FROMNWSSTG(FROMSTG) ASP(3)
This command creates a network server storage space called STGSPACE4 with a size and format the same as FROMSTG and copies the contents into STGSPACE4. It will be created in user auxiliary storage pool (ASP) 3.

---

### Error messages

*ESCAPE Messages*

**CPFA42D**

Storage space &1 not created.
Create Output Queue (CRTOUTQ)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Create Output Queue (CRTOUTQ) command creates a new output queue for spooled files. An entry is placed on the output queue for each spooled file. The order in which the files are written to the output device is determined by the output priority of the spooled file and the value specified on the Order of files on queue (SEQ) parameter.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>OUTQ</td>
<td>Output queue</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Output queue</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>MAXPAGES</td>
<td>Maximum spooled file size</td>
<td>Single values: *NONE, Other values (up to 5 repetitions): Element list</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Element 1: Number of pages</td>
<td>Integer</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Starting time</td>
<td>Time</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 3: Ending time</td>
<td>Time</td>
<td></td>
</tr>
<tr>
<td>SEQ</td>
<td>Order of files on queue</td>
<td>*FIFO, *JOBNBR</td>
<td>Optional</td>
</tr>
<tr>
<td>RMTPRQT</td>
<td>Remote printer queue</td>
<td>Character value, *USER, *SYSTEM</td>
<td>Optional</td>
</tr>
<tr>
<td>AUTOSTRWR</td>
<td>Writers to autostart</td>
<td>1-10, *NONE</td>
<td>Optional</td>
</tr>
<tr>
<td>MSGQ</td>
<td>Queue for writer messages</td>
<td>Qualified object name</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Queue for writer messages</td>
<td>Name, QSYSOPR</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>CNNTYPE</td>
<td>Connection type</td>
<td>Character value, *SNA, *IP, *USRDFN</td>
<td>Optional</td>
</tr>
<tr>
<td>TRANSFORM</td>
<td>Host print transform</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
<tr>
<td>USRDTATFM</td>
<td>User data transform</td>
<td>Single values: *NONE, Other values: Qualified object name</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: User data transform</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>MFRTYPMDL</td>
<td>Manufacturer type and model</td>
<td>Character value, *WSCST</td>
<td>Optional</td>
</tr>
<tr>
<td>WSCST</td>
<td>Workstation customizing object</td>
<td>Single values: *NONE, Other values: Qualified object name</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Workstation customizing object</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
</tr>
</tbody>
</table>
### Output queue (OUTQ)

Specifies the output queue being created.

This is a required parameter.

**Qualifier 1: Output queue**

*name* Specify the name of the output queue being created.

**Qualifier 2: Library**

---

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>IMGCFG</td>
<td>Image configuration</td>
<td><em>NONE</em></td>
<td>Optional</td>
</tr>
<tr>
<td>INTNETADR</td>
<td>Internet address</td>
<td><em>NONE</em></td>
<td>Optional</td>
</tr>
<tr>
<td>CLASS</td>
<td>VM/MVS class</td>
<td>A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U, V, W, X, Y, Z, 0, 1, 2, 3, 4, 5, 6, 7, 8, 9</td>
<td>Optional</td>
</tr>
<tr>
<td>PCB</td>
<td>Forms Control Buffer</td>
<td><em>NONE</em>, <em>USRDTA</em>, <em>PRTF</em></td>
<td>Optional</td>
</tr>
<tr>
<td>DESTOPT</td>
<td>Destination options</td>
<td><em>NONE</em>, <em>USRDFNTXT</em>, <em>NOWAIT</em></td>
<td>Optional</td>
</tr>
<tr>
<td>SEPPAGE</td>
<td>Print separator page</td>
<td><em>YES</em>, <em>NO</em></td>
<td>Optional</td>
</tr>
<tr>
<td>USRDFNOPT</td>
<td>User defined option</td>
<td>Single values: <em>NONE</em></td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Other values (up to 4 repetitions): <em>Character value</em></td>
<td></td>
</tr>
<tr>
<td>USRDFNOBJ</td>
<td>User defined object</td>
<td>Single values: <em>NONE</em></td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Other values: <em>Element list</em></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 1: Object</td>
<td>Qualified object name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Object</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td><em>LIBL</em>, <em>CURLIB</em></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Object</td>
<td><em>DTAARA</em>, <em>DTAQ</em>, <em>FILE</em>, <em>PSFCFG</em>, <em>USRIDX</em>, <em>USRQ</em>, <em>USRSPC</em></td>
<td></td>
</tr>
<tr>
<td>USRDRVPGM</td>
<td>User driver program</td>
<td>Single values: <em>NONE</em></td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Other values: <em>Qualified object name</em></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: User driver program</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td><em>LIBL</em>, <em>CURLIB</em></td>
<td></td>
</tr>
<tr>
<td>SPLFASP</td>
<td>Spooled file ASP</td>
<td><em>SYSTEM</em>, <em>OUTQASP</em></td>
<td>Optional</td>
</tr>
<tr>
<td>TEXT</td>
<td>Text ‘description’</td>
<td><em>BLANK</em></td>
<td>Optional</td>
</tr>
<tr>
<td>DSPDTA</td>
<td>Display any file</td>
<td><em>NO</em>, <em>YES</em>, <em>OWNER</em></td>
<td>Optional, Positional 2</td>
</tr>
<tr>
<td>JOBSEP</td>
<td>Job separators</td>
<td>0-9, 0, <em>MSG</em></td>
<td>Optional, Positional 3</td>
</tr>
<tr>
<td>OPRCTL</td>
<td>Operator controlled</td>
<td><em>YES</em>, <em>NO</em></td>
<td>Optional, Positional 4</td>
</tr>
<tr>
<td>DTAQ</td>
<td>Data queue</td>
<td>Single values: <em>NONE</em></td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Other values: <em>Qualified object name</em></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Data queue</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td><em>LIBL</em>, <em>CURLIB</em></td>
<td></td>
</tr>
<tr>
<td>AUTCHK</td>
<td>Authority to check</td>
<td><em>OWNER</em>, <em>DTAAUT</em></td>
<td>Optional</td>
</tr>
<tr>
<td>AUT</td>
<td>Authority</td>
<td>Name, <em>USE</em>, <em>ALL</em>, <em>CHANGE</em>, <em>EXCLUDE</em>, <em>LIBCRTAUT</em></td>
<td>Optional</td>
</tr>
</tbody>
</table>
**CURLIB**

The current library for the job is used to locate the output queue. If no current library entry exists in the library list, QGPL is used.

*name*

Specify the name of the library where the output queue is to be located.

**Note:** The temporary library QTEMP is not a valid library name. Output queues must be in permanent libraries.

---

**Maximum spooled file size (MAXPAGES)**

Specifies the maximum spooled file size in pages that will be allowed to print between a starting and ending time. If a spooled file exceeds the page limit it will be deferred (DFR status) until the ending time expires. For files where the exact number of pages is not known, the estimated number of pages is used.

(You can use the Work with Spooled File Attributes (WRKSPFLA) command to find out the estimated number of pages.) Time must be specified in hhmmss format, on a 24 hour clock.

**Single values**

*NONE*

There is no limit on the size of spooled files allowed to print from this output queue.

**Other values (up to 5 repetitions)**

**Element 1: Number of pages**

*integer*

Specify the largest spooled file, in pages, that is allowed to print.

**Element 2: Starting time**

*time*

Specify the time of day that the maximum spooled file size limit is to start.

**Element 3: Ending time**

*time*

Specify the time of day that the maximum spooled file size limit is to end.

---

**Order of files on queue (SEQ)**

Specifies the order of the spooled files on the output queue.

*FIFO*

The queue is first-in first-out within priority for each file. That is, new spooled files are placed after all other entries on the queue of the same priority.

*JOBNBR*

The queue entries for spooled files are sorted in priority sequence using the job number (actually, the date and time that the job entered the system is used) of the job that created the spooled file.

---

**Remote system (RMTSYS)**

Specifies the remote system to send files to when a remote writer is started (using the STRRMTWTR command) to the output queue. This is referred to as the "address" by SNADS, and the "host" by TCP/IP.
**NONE**

The output queue is used only for local printing. The STRRMTWTR command cannot be used when this output queue is specified on the OUTQ parameter.

**PASTHR**

The system a user passed through (using the STRPASTHR command) is used when sending spooled files created by the user job. If a spooled file was not created by a job that had passed through from another system, the spooled file will be held (HLD status).

**INTNETADR**

The INTNETADR parameter is used to identify the system when a remote writer is started to the output queue. If you have a host table or a domain name server on your TCP/IP network, you can use the remote-system-name instead of this parameter.

**Note:** This value is valid only when *IP has been specified for the CNNTYPE parameter.

**NWSA**

The RMTPRTQ parameter is used to identify the system when a remote writer is started to the output queue. This value is valid only when *NDS has been specified for the DESTTYPE parameter.

`remote-system-name`

Specify a name for the remote system. Only the first 8 characters will be used when the connection type (CNNTYPE parameter) is specified as *SNA. If the name of the remote system needs to be lower case, the name must be enclosed in apostrophes. If you do not use apostrophes, the operating system changes the name to upper case.

---

**Remote printer queue (RMTPRTQ)**

Specifies the printer queue on the remote system (RMTSYS parameter) to which the remote writer sends spooled files.

**USER**

The user profile that created the spooled file determines the user ID on the remote system. This value is valid only when the connection type (CNNTYPE parameter) is specified as *SNA or *USRDFN.

**SYSTEM**

The default system printer on the remote system is used to determine the printer queue. For a remote iSeries family system, the output queue associated with the printer device specified in the QPRDEV system value is used as the printer queue.

**Note:** This value is valid only when the connection type (CNNTYPE parameter) is specified as *SNA or *USRDFN and the DESTTYPE is *OS400 or *S390.

`name`

Specifies the name for the printer queue on the remote system. For remote systems that are iSeries family systems, this is the name of an output queue on which the spooled file is created. If the name of the remote system needs to be lower case, the name must be enclosed in apostrophes. If you do not use apostrophes, the operating system changes the name to upper case.

For destination systems that are not iSeries family systems, this name is system-dependent, and can be either the actual name of the device or the name of a printer queue.

This output queue is usually specified as library name/output queue name. If a library name qualifier is not specified, the value *LBL is used as the default.
Writers to autostart (AUTOSTRWTR)

Specifies the number of remote writers that will be started automatically by the system. For user created output queues with the remote system (RMTSYS parameter) specified as *NONE, this parameter will be ignored.

*NONE
There will be no writers auto-started by the system to this output queue.

1-10 Specify the number of writers to be auto-started to this output queue.

Queue for writer messages (MSGQ)

Specifies the message queue to which messages are sent when created by the remote writer started to this output queue.

Qualifier 1: Queue for writer messages

QSYSOPR
Messages are sent to the QSYSOPR message queue.

name Specify the name of the message queue to which messages created by the remote writer are sent.

Qualifier 2: Library

*LIBL All libraries in the library list for the current thread are searched until the first match is found.

*CURLIB The current library for the thread is searched. If no library is specified as the current library for the thread, the QGPL library is searched.

name Specify the name of the library to be searched.

Connection type (CNNTYPE)

Specifies the type of connection with the remote system.

*SNA The spooled files are sent using SNADS. This is similar to the Send Network Spooled File (SNDNETSPLF) command and requires that SNADS be configured.

*IP The spooled files will be sent using TCP/IP. This is similar to the Send TCP/IP Spooled File (SNDTCPSPFLF) command and requires that the TCP/IP product be installed.

*USRDFN
The spooled files are sent using a user-defined connection.

Destination type (DESTTYPE)

Specifies the type of the remote system (RMTSYS parameter). This parameter, along with the type of data contained in the spooled file (DEVTYPE parameter on the CRTPTRTF command), is used by a remote writer to determine the format used to send the spooled file. The spooled file will be held by the remote writer if the type of data in the spooled file is not supported by the system.
The spooled files are to be sent to an iSeries family system running OS/400 V3R1M0 or later, or i5/OS, when the connection type (CNNTYPE) has been specified as *SNA. This value can be specified for all releases which support TCP/IP (V2R3 and later) when CNNTYPE is *IP or when CNNTYPE is *USRDFN.

Note: This value should be specified when possible, to allow the greatest flexibility when selecting values for other parameters.

The spooled files are sent to an iSeries family system running OS/400 versions prior to V3R1M0. This value is valid only when CNNTYPE is *SNA.

The spooled files are sent to a System/390 system. This value is valid only when CNNTYPE is *SNA or when CNNTYPE is *USRDFN.

The spooled files are sent to a personal computer running the PSF/2 product. This value is valid only when the CNNTYPE is *IP.

The spooled files are to be sent to NETWARE4. This value is valid only when the CNNTYPE is *IP or *USRDFN.

The spooled files are sent to a system not matching any of the other special values. This includes iSeries family systems running OS/400 version 1, as well as System/36 and System/38 systems.

Transform SCS to ASCII (TRANSFORM)

Specifies whether or not to make use of the host print transform function to transform a spooled file of device type *SCS into ASCII data when the file is sent to a remote printer queue.

Note: This parameter is not valid when the CNNTYPE is specified as *SNA or *NONE.

*YES The SCS data streams are transformed.

*NO The SCS data streams are not transformed.

Data transform program (USRDTATFM)

Specifies the user-defined data program that is used to transform the spooled file data.

Note: This parameter is valid only when RMTSYS is not *NONE.

Single values

*NONE No user-defined data transform program name is specified.

Qualifier 1: User data transform

ame Specify the name of the data transform program to be used by the driver program.

Qualifier 2: Library

*LIBL All libraries in the library list for the current thread are searched until the first match is found.
*CURLIB

The current library for the thread is searched. If no library is specified as the current library for the thread, the QGPL library is searched.

name Specify the name of the library to be searched.

Manufacturer type and model (MFRTYPMDL)

Specifies the manufacturer, type, and model for a printer using the host print transform function or user data transform program. This parameter is only prompted when TRANSFORM(*YES) is specified or a user data transform program is specified.

If *WSCSTxxx is specified for MFRTYPMDL, a workstation customizing object must be specified.

*IBM42011

The IBM 4201-1 Proprinter is used.

*WSCST

The value of the WSCST parameter is used.

character-value

Specify the manufacturer, type, and model for a printer using the host print transform function.

Manufacturer Type and Model Table

<table>
<thead>
<tr>
<th>Manufacturer Type and Model Table</th>
<th>ASCII</th>
<th>Product Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>IBM2380</strong></td>
<td>1545</td>
<td>2380 Personal Printer Series II</td>
</tr>
<tr>
<td></td>
<td>1546</td>
<td>2380 Plus Printer</td>
</tr>
<tr>
<td><strong>IBM2381</strong></td>
<td>1545</td>
<td>2381 Personal Printer Series II</td>
</tr>
<tr>
<td></td>
<td>1546</td>
<td>2381 Plus Printer</td>
</tr>
<tr>
<td><strong>IBM2390</strong></td>
<td>1545</td>
<td>2390 Personal Printer Series II</td>
</tr>
<tr>
<td></td>
<td>1546</td>
<td>2390 Plus Printer</td>
</tr>
<tr>
<td><strong>IBM2391</strong></td>
<td>1545</td>
<td>2391 Personal Printer Series II</td>
</tr>
<tr>
<td></td>
<td>1546</td>
<td>2391 Plus Printer</td>
</tr>
<tr>
<td><strong>IBM3112</strong></td>
<td>1545</td>
<td>3112 Page Printer</td>
</tr>
<tr>
<td><strong>IBM3116</strong></td>
<td>1545</td>
<td>3116 Page Printer</td>
</tr>
<tr>
<td><strong>IBM3130</strong></td>
<td>1545</td>
<td>3130 Advanced Function Printer</td>
</tr>
<tr>
<td><strong>IBM3812</strong></td>
<td>1545</td>
<td>3812 Pageprinter</td>
</tr>
<tr>
<td></td>
<td>1546</td>
<td>3816 Pageprinter</td>
</tr>
<tr>
<td><strong>IBM3912HP</strong></td>
<td>1545</td>
<td>3912 Page Printer (HP Mode)</td>
</tr>
<tr>
<td><strong>IBM3916HP</strong></td>
<td>1545</td>
<td>3916 Page Printer (HP Mode)</td>
</tr>
<tr>
<td><strong>IBM39302</strong></td>
<td>1545</td>
<td>3930-025 Page Printer</td>
</tr>
<tr>
<td></td>
<td>1546</td>
<td>3930-020 Page Printer</td>
</tr>
<tr>
<td><strong>IBM39303</strong></td>
<td>1545</td>
<td>3930-035 Page Printer</td>
</tr>
<tr>
<td></td>
<td>1546</td>
<td>3930-030 Page Printer</td>
</tr>
<tr>
<td><strong>IBM4019</strong></td>
<td>1545</td>
<td>4019 LaserPrinter</td>
</tr>
<tr>
<td></td>
<td>1546</td>
<td>4019E LaserPrinter E</td>
</tr>
<tr>
<td><strong>IBM4019HP</strong></td>
<td>1545</td>
<td>4019 LaserPrinter (HP Mode)</td>
</tr>
<tr>
<td></td>
<td>1546</td>
<td>4019E LaserPrinter E (HP Mode)</td>
</tr>
<tr>
<td><strong>IBM4029</strong></td>
<td>1545</td>
<td>4029-010 LaserPrinter 5E</td>
</tr>
<tr>
<td></td>
<td>1546</td>
<td>4029-020 LaserPrinter 6</td>
</tr>
<tr>
<td></td>
<td>1546</td>
<td>4029-030 LaserPrinter 10</td>
</tr>
<tr>
<td></td>
<td>1547</td>
<td>4029-040 LaserPrinter 10L</td>
</tr>
<tr>
<td><strong>IBM4029HP</strong></td>
<td>1545</td>
<td>4029-010 LaserPrinter 5E (HP Mode)</td>
</tr>
<tr>
<td></td>
<td>1546</td>
<td>4029-020 LaserPrinter 6 (HP Mode)</td>
</tr>
<tr>
<td></td>
<td>1546</td>
<td>4029-030 LaserPrinter 10 (HP Mode)</td>
</tr>
<tr>
<td></td>
<td>1547</td>
<td>4029-040 LaserPrinter 10L (HP Mode)</td>
</tr>
<tr>
<td><strong>IBM4037</strong></td>
<td>1545</td>
<td>4037 5E Printer</td>
</tr>
<tr>
<td><strong>IBM4039HP</strong></td>
<td>1545</td>
<td>4039 LaserPrinter 10D (HP Mode)</td>
</tr>
<tr>
<td></td>
<td>1546</td>
<td>4039 LaserPrinter 10D Plus (HP Mode)</td>
</tr>
<tr>
<td></td>
<td>1546</td>
<td>4039 LaserPrinter 10R (HP Mode)</td>
</tr>
</tbody>
</table>
IBM 4039 LaserPrinter 10R Plus (HP Mode)
IBM 4039 LaserPrinter 12R (HP Mode)
IBM 4039 LaserPrinter 12R Plus (HP Mode)
IBM 4039 LaserPrinter 12L (HP Mode)
IBM 4039 LaserPrinter 12L Plus (HP Mode)
IBM 4039 LaserPrinter 16L (HP Mode)
IBM 4039 LaserPrinter 16L Plus (HP Mode)

*IBM4070 IBM 4070 1J (IBM Mode)
*IBM4070EP IBM 4070 1J (Epson Mode)
*IBM4072 IBM 4072 ExecJet
*IBM4076 IBM 4076 ExecJet II Printer (HP Mode)
*IBM42011 IBM 4201-1 Proprinter
*IBM42012 IBM 4201-2 Proprinter II
*IBM42013 IBM 4201-3 Proprinter III
*IBM42021 IBM 4202-1 Proprinter XL
*IBM42022 IBM 4202-2 Proprinter II XL
*IBM42023 IBM 4202-3 Proprinter III XL
*IBM42071 IBM 4207-1 Proprinter X24
*IBM42072 IBM 4207-2 Proprinter X24E
*IBM42081 IBM 4208-1 Proprinter XL24
*IBM42082 IBM 4208-2 Proprinter XL24E
*IBM4212 IBM 4212 Proprinter 24P
*IBM4216 IBM 4216-10 Personal Pageprinter
*IBM4226 IBM 4226-302 Printer
*IBM4230 IBM 4230-4S3 Printer (IBM Mode)
*IBM4244ASF IBM 4244 Printer (single tractor feed and ASF)
*IBM4244DUAL IBM 4244 Printer (dual tractor feeds)
*IBM4244MAN IBM 4244 Printer (single form feed or manual selection)
*IBM4247ASF IBM 4247 Printer (single tractor feed and ASF)
*IBM4247DUAL IBM 4247 Printer (dual tractor feeds)
*IBM4247MAN IBM 4247 Printer (single form feed or manual selection)
*IBM4308 IBM Infoprint Color 8
*IBM4312 IBM Network Printer 12
*IBM4317 IBM Network Printer 17
*IBM4320 IBM Infoprint 20
*IBM4322 IBM Infoprint 21
*IBM4324 IBM Network Printer 24
*IBM4332 IBM Infoprint 32
*IBM4340 IBM Infoprint 40
*IBM47121 IBM 4712-1 Transaction Printer
*IBM47122 IBM 4712-2 Transaction Printer
*IBM47221 IBM 4722-1 Document Printer
*IBM47222 IBM 4722-2 Document Printer
*IBM4770 IBM 4770 InkJet Transaction Printer
*IBM4912 IBM Infoprint 12
*IBM5152 IBM 5152 Graphics Printer
*IBM5201 IBM 5201-2 Quietwriter
*IBM5202 IBM 5202-1 Quietwriter III
*IBM5204 IBM 5204-1 Quickwriter
*IBM5216 IBM 5216 Wheelprinter
*IBM5575 IBM 5579-H02 Printer
IBM 5579-K02 Printer
IBM 5577-T02 Printer
IBM 5579-S02 Printer
IBM 5577-K02 Printer
IBM 5577-J02 Printer
IBM 5577-G02 Printer
IBM 5577-H02 Printer
IBM 5577-F02 Printer
IBM 5577-B02 Printer
IBM 5575-H02 Printer
IBM 5575-F02 Printer (with SBCS Cartridge)
IBM 5575-B02 Printer (with SBCS Cartridge)
IBM 5573-K02 Printer
IBM 5573-J02 Printer
IBM 5573-H02 Printer
IBM 5573-G02 Printer
IBM 5572-B02 Printer
IBM 5417-011 Printer
IBM 5407-011 Printer
IBM 5327-011 Printer
IBM 4208-502 Printer

*IBM6400 IBM 6400 Printers (IBM Mode)
*IBM6400EP IBM 6400 Printers (Epson Mode)
*IBM6404 IBM 6404 Printers (IBM Mode)
*IBM6404EP IBM 6404 Printers (Epson Mode)
*IBM6408 IBM 6408-A00 Printer (IBM Mode)
*IBM6408EP IBM 6408-A00 Printer (Epson Mode)
*IBM6412 IBM 6412-A00 Printer (IBM Mode)
*IBM6412EP IBM 6412-A00 Printer (Epson Mode)

*IBMPAGES IBM 5589-H01 Printer
IB MPAGES IBM 5588-H02 Printer
IBM 5587-H01 Printer
IBM 5586-H02 Printer
IBM 5585-H01 Printer
IBM 5584-K02 Printer
IBM 5584-H02 Printer
IBM 5584-G02 Printer

*IBMPAGESNPB Same as *IBMPAGES, but without text positioning adjustments for a no-print border

*IBMPAGES300 IBM Network Printer 12 (with PAGES feature)
IBM Network Printer 17 (with PAGES feature)
IBM Network Printer 24 (with PAGES feature)
IBM Infoprint 20 (with PAGES feature)
IBM Infoprint 32 (with PAGES feature)
IBM Infoprint 40 (with PAGES feature)

*IBMPAGES300NPB Same as *IBMPAGES300, but without text positioning adjustments for a no-print border

*INFOPRINT8C IBM Infoprint Color 8
*INFOPRINT12 IBM Infoprint 12
*INFOPRINT20 IBM Infoprint 20
*INFOPRINT21 IBM Infoprint 21
*INFOPRINT32 IBM Infoprint 32
*INFOPRINT40 IBM Infoprint 40
*INFOPRINT70 IBM Infoprint 70
*INFOPRINT85 IBM Infoprint 2085
*INFOPRINT105 IBM Infoprint 2105
*INFOPRINT1116 IBM Infoprint 1116
*INFOPRINT1120 IBM Infoprint 1120
*INFOPRINT1125 IBM Infoprint 1125
*INFOPRINT1130 IBM Infoprint 1130
*INFOPRINT1140 IBM Infoprint 1140
*INFOPRINT1145 IBM Infoprint 1145

Create Output Queue (CRTOUTQ)  101
*INFOPRINT1220C  IBM Infoprint Color 1220
*INFOPRINT1222  IBM Infoprint 1222
*INFOPRINT1226  IBM Infoprint 1226
*INFOPRINT1228C IBM Infoprint Color 1228
*INFOPRINT1312  IBM Infoprint 1312
*INFOPRINT1332  IBM Infoprint 1332
*INFOPRINT1334C IBM Infoprint Color 1334
*INFOPRINT1352  IBM Infoprint 1352
*INFOPRINT1354C IBM Infoprint Color 1354
*INFOPRINT1357C IBM Infoprint Color 1357
*INFOPRINT1372  IBM Infoprint 1372
*INFOPRINT1400C IBM Infoprint Color 14xx Series Printer
*INFOPRINT1410  IBM Infoprint 1410 MFP
*INFOPRINT1412  IBM Infoprint 1412
*INFOPRINT1422  IBM Infoprint 1422
*INFOPRINT2000  IBM Infoprint 2000
*INFOPRINT2085  IBM Infoprint 2085
*INFOPRINT2105  IBM Infoprint 2105
*INFOPRINT2105ES IBM Infoprint 2105ES
*INFOPRINT2600  IBM Infoprint 2600
*INFOPRINT2650  IBM Infoprint 2650
*INFOPRINT2705  IBM Infoprint 2705
*INFOPRINT2706  IBM Infoprint 2706
*INFOPRINT2710  IBM Infoprint 2710
*INFOPRINT2760  IBM Infoprint 2760
*INFOPRINT2785  IBM Infoprint 2785
*INFOPRINT2790  IBM Infoprint 2790
*CANLIPS3  Canon LIPS3 DBCS Printers
*CANLIPS3NPB Same as *CANLIPS3, but without text
positioning adjustments for a no-print border
*CPQPM15  COMPAQ PageMark 15 (HP Mode)
*CPQPM20  COMPAQ PageMark 20 (HP Mode)
*EPAP2250  Epson ActionPrinter 2250
*EPAP3250  Epson ActionPrinter 3250
*EPAP5000  Epson ActionPrinter 5000
*EPAP5600  Epson ActionPrinter 5500
*EPDFX5000  Epson DFX-5000
*EPDFX8000  Epson DFX-8000
*EPFX5000  Epson FX-5000
*EPFX5600  Epson FX-5060
*EPFX5700  Epson FX-5700
*EPFX6000  Epson FX-6000
*EPFX6200  Epson FX-6200
*EPFX8000  Epson FX-8000
*EPFX8100  Epson FX-8100
*EPFX8500  Epson FX-8500
*EPL5000  Epson LQ-5000
*EPL5600  Epson LQ-5600
*EPL5700  Epson LQ-5700
*EPL5800  Epson LQ-5800
*EPL5900  Epson LQ-5900
*EPL8000  Epson LQ-8000
*EPL8100  Epson LQ-8100
*EPL8600  Epson LQ-8600
*EPL8700  Epson LQ-8700
*EPL8900  Epson LQ-8900
*EPLX5000  Epson LX-5000
*EPLX5600  Epson LX-5600
*EPLX5700  Epson LX-5700
*EPLX5800  Epson LX-5800
*EPLX5900  Epson LX-5900
*EPLX6000  Epson LX-6000
*EPLX6100  Epson LX-6100
*EPLX6200  Epson LX-6200
*EPLX8000  Epson LX-8000
*EPLX8100  Epson LX-8100
*EPLX8200  Epson LX-8200
*EPSQ5000  Epson SQ-5000
*EPSQ5100  Epson SQ-5100
*EPSQ5200  Epson SQ-5200
*EPSQ5500  Epson SQ-5500
*EPSQ6000  Epson SQ-6000
*EPSQ6100  Epson SQ-6100
*EPSQ6200  Epson SQ-6200
*EPSQ8000  Epson SQ-8000
*EPSQ8100  Epson SQ-8100
*EPSQ8200  Epson SQ-8200
*ESCRPDBCS  Epson ESC/P DBCS Printers
*HP5  HP LaserJet Series II
*HP510  HP LaserJet IID
*HP511  HP LaserJet IIP
*HP550  HP LaserJet III
*HP551  HP LaserJet IID
*HP551P  HP LaserJet IIP
*HP551P  HP LaserJet III
*HP55I  HP LaserJet 4
*HP55I  HP LaserJet 5 series
*HP55I  HP LaserJet 55I
*HP56  HP LaserJet 6 series
*HP56  HP DeskJet 310
*HP56  HP DeskJet 320
*HP500  HP DeskJet 500
*HP520  HP DeskJet 520
*HP540  HP DeskJet 540
*HP550C HP DeskJet 550C
*HP560C HP DeskJet 560C
*HP1100 HP LaserJet 1100 series
*HP1200C HP DeskJet 1200C
*HP1600C HP LaserJet 1600C
*HP4000 HP LaserJet 4000 series
*HP5000 HP LaserJet 5000 series
*HP8000 HP LaserJet 8000 series
*HPCOLORLJ HP Color LaserJet 5
*HPDBCS HP LaserJet-compatible printers for Double Byte Character Set (DBCS) input
*HPPAINT HP PaintJet
HP PaintJet XL
*LEX2380 Lexmark Forms Printer 2380 Plus
*LEX2381 Lexmark Forms Printer 2381 Plus
*LEX2390 Lexmark Forms Printer 2390 Plus
*LEX2391 Lexmark Forms Printer 2391 Plus
*LEX4227 Lexmark 4227 Forms Printer
*LEXMARKC Lexmark C Series Printer
*LEXMARKC510 Lexmark C510 Color Printer
*LEXMARKC750 Lexmark C750 Color Printer
*LEXMARKC752 Lexmark C752 Color Printer
*LEXMARKC910 Lexmark C910 Color Printer
*LEXMARKC912 Lexmark C912 Color Printer
*LEXMARKE Lexmark E Series Printer
*LEXMARKE322 Lexmark E322 Printer
*LEXMARKE323 Lexmark E323 Printer
*LEXMARKE330 Lexmark E330 Printer
Lexmark E332n Printer
*LEXMARKT Lexmark T Series Printer
*LEXMARKT420 Lexmark T420 Printer
*LEXMARKT520 Lexmark T520 Printer
*LEXMARKT522 Lexmark T522 Printer
*LEXMARKT620 Lexmark T620 Printer
*LEXMARKT622 Lexmark T622 Printer
*LEXMARKT630 Lexmark T630 Printer
*LEXMARKT632 Lexmark T632 Printer
*LEXMARKT634 Lexmark T634 Printer
*LEXMARKW Lexmark W Series Printer
*LEXMARKWB12 Lexmark WB12 Printer
*LEXMARKWB20 Lexmark WB20 Printer
*LEXMARKX422 Lexmark X422 MFP
*LEXOPTRA Lexmark Optra Family (HP Mode)
*LEXOPTRAC Lexmark Optra C Color Printer
*LEXOPTRAN Lexmark Optra N Printer
*LEXOPTRAS Lexmark Optra S Printer family
*LEXOPTRASC Lexmark Optra SC Color Printer
Lexmark Optra Color 1200 Printer
*LEXOPTRAT Lexmark Optra T Printer series
*LEXOPTRAW Lexmark Optra W Printer series
*NECP2 NEC P2 Pinwriter
*NECP2200 NEC P2200 Pinwriter
*NECP2200XE NEC P2200 XE Pinwriter
*NECP5200 NEC P5200 Pinwriter
*NECP5300 NEC P5300 Pinwriter
*NECP6200 NEC P6200 Pinwriter
*NECP6300 NEC P6300 Pinwriter
*NECP CPR201 NEC PC-PR101 DBCS Printer
NEC PC-PR201 DBCS Printer
*NONE Printer supports page-descriptor language generated by the CVTIMG API.
NOTE: Spoolfiles with device type of *SCS or *AFPDS cannot be processed by the Host Print Transform function for

Create Output Queue (CRTOUTQ) 103
these printers.

*OKI184IBM Okidata Microline 184 Turbo (IBM Mode)
*OKI320IBM Okidata Microline 320 (IBM Mode)
*OKI321IBM Okidata Microline 321 (IBM Mode)
*OKI390IBM Okidata Microline 390 Plus (IBM Mode)
*OKI391IBM Okidata Microline 391 Plus (IBM Mode)
*OKI590IBM Okidata Microline 590 (IBM Mode)
*OKI591IBM Okidata Microline 591 (IBM Mode)
*OKI400 Okidata OL400 LED Page Printer
*OKI800 Okidata OL800 LED Page Printer
*OKI810 Okidata OL810 LED Page Printer
*OKI820 Okidata OL820 LED Page Printer
*OKI3410 Okidata Pacemark 3410
*PAN1123EP Panasonic KX-P1123 (Epson Mode)
*PAN1124EP Panasonic KX-P1124 (Epson Mode)
*PAN1124IEP Panasonic KX-P1124i (Epson Mode)
*PAN1180EP Panasonic KX-P1180 (Epson Mode)
*PAN1180IEP Panasonic KX-P1180i (Epson Mode)
*PAN191IEP Panasonic KX-P1191 (Epson Mode)
*PAN1624EP Panasonic KX-P1624 (Epson Mode)
*PAN1654EP Panasonic KX-P1654 (Epson Mode)
*PAN1695EP Panasonic KX-P1695 (Epson Mode)
*PAN2123EP Panasonic KX-P2123 (Epson Mode)
*PAN2124EP Panasonic KX-P2124 (Epson Mode)
*PAN2180EP Panasonic KX-P2180 (Epson Mode)
*PAN2624EP Panasonic KX-P2624 (Epson Mode)
*PAN4410HP Panasonic KX-P4410 (HP Mode)
*PAN4420HP Panasonic KX-P4420 (HP Mode)
*PAN4430HP Panasonic KX-P4430 (HP Mode)
*PAN4451HP Panasonic KX-P4451 (HP Mode)
*PANASONIC2310 Panasonic DP-2310 Printer
*PANASONIC3010 Panasonic DP-3010 Printer
*PANASONIC3510 Panasonic DP-3510 Printer
*PANASONIC3520 Panasonic DP-3520 Printer
*PANASONIC4510 Panasonic DP-4510 Printer
*PANASONIC4520 Panasonic DP-4520 Printer
*PANASONIC6010 Panasonic DP-6010 Printer
*PANASONIC6020 Panasonic DP-6020 Printer
*RICOH1515 Ricoh Aficio 1515 Printer Series
*RICOH2015 Ricoh Aficio 2015 Printer Series
*RICOH2018 Ricoh Aficio 2018 Printer Series
*RICOH2022 Ricoh Aficio 2022 Printer Series
*RICOH2027 Ricoh Aficio 2027 Printer Series
*RICOH2032 Ricoh Aficio 2032 Printer Series
*RICOH2035 Ricoh Aficio 2035 Printer Series
*RICOH2045 Ricoh Aficio 2045 Printer Series
*RICOHAP400 Ricoh Aficio AP400 Printer Series
*RICOHAP600N Ricoh Aficio AP600N Printer Series
*RICOHAP900 Ricoh Aficio AP900 Printer Series
*RICOHAP3200 Ricoh Aficio AP3200 Printer Series
*RICOHAP4510 Ricoh Aficio AP4510 Printer Series
*RICOHCL2000 Ricoh Aficio CL2000 Color Printer Series
*RICOHCL3100 Ricoh Aficio CL3000e Color Printer Series
*RICOHCL3100N Ricoh Aficio CL3100N Color Printer Series
*RICOHCL4000 Ricoh Aficio CL4000 Color Printer Series
*RICOHCL5000 Ricoh Aficio CL5000 Color Printer Series
*RICOHCL7000 Ricoh Aficio CL7000 Color Printer Series
*RICOHCL7100 Ricoh Aficio CL7100 Color Printer Series
Workstation customizing object (WSCST)

Specifies an object that consists of a table of attributes used to customize a given ASCII device, such as a workstation or printer. Character presentation, font specifications and control key sequences are examples of characteristics that can be customized.

This parameter is only prompted when TRANSFORM(*YES) is specified, or when a user data transform program is used.

Single values

*NONE

No workstation customizing object is specified.

Qualifier 1: Workstation customizing object

name Specify the name of a workstation customizing object, which has been created with the Create Work Station Customizing Object (CRTWSCST) command.

Qualifier 2: Library

*LIBL All libraries in the library list for the current thread are searched until the first match is found.

*CURLIB

The current library for the thread is searched. If no library is specified as the current library for the thread, the QGPL library is searched.
**name** Specify the name of the library to be searched.

---

**Image configuration (IMGCFG)**

Specifies the image configuration for this output queue. An image configuration object provides transform services for a variety of image and print datastream formats. This parameter is only used with remote writers.

See the Image Configuration Object (IMGCFG Parameter) table below for a list of the image configuration objects provided.

See the Recommended Image Configuration Objects by Printer table below for the suggested IMGCFG object for many popular printers.

*NONE*

No image configuration specified.

**character-value**

Specify the image configuration to be used for this output queue.

---

**Image Configuration Object Table**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>IMGA01</em></td>
<td>PCL 300-dpi printer</td>
</tr>
<tr>
<td><em>IMGA02</em></td>
<td>PCL 600-dpi printer</td>
</tr>
<tr>
<td><em>IMGA03</em></td>
<td>PCL 1200-dpi printer</td>
</tr>
<tr>
<td><em>IMGA04</em></td>
<td>PCL 300-dpi color printer</td>
</tr>
<tr>
<td><em>IMGA05</em></td>
<td>PCL 600-dpi color printer</td>
</tr>
<tr>
<td><em>IMGA06</em></td>
<td>PCL 1200-dpi color printer</td>
</tr>
<tr>
<td><em>IMGA07</em></td>
<td>PCL 75-dpi printer (No compression)</td>
</tr>
<tr>
<td><em>IMGA08</em></td>
<td>PCL 600-dpi color printer with larger no-print border</td>
</tr>
<tr>
<td><em>IMGA09</em></td>
<td>PCL 300-dpi printer (No compression)</td>
</tr>
<tr>
<td><em>IMGB01</em></td>
<td>Postscript 300-dpi printer</td>
</tr>
<tr>
<td><em>IMGB02</em></td>
<td>Postscript 600-dpi printer</td>
</tr>
<tr>
<td><em>IMGB03</em></td>
<td>Postscript 1200-dpi printer</td>
</tr>
<tr>
<td><em>IMGB04</em></td>
<td>Postscript 300-dpi color printer</td>
</tr>
<tr>
<td><em>IMGB05</em></td>
<td>Postscript 600-dpi color printer</td>
</tr>
<tr>
<td><em>IMGB06</em></td>
<td>Postscript 1200-dpi color printer</td>
</tr>
<tr>
<td><em>IMGB07</em></td>
<td>Postscript 600x300-dpi color printer</td>
</tr>
<tr>
<td><em>IMGB08</em></td>
<td>Postscript 1200x300-dpi color printer</td>
</tr>
<tr>
<td><em>IMGB09</em></td>
<td>Postscript 360-dpi color printer</td>
</tr>
<tr>
<td><em>IMGB10</em></td>
<td>Postscript 720-dpi color printer</td>
</tr>
<tr>
<td><em>IMGB11</em></td>
<td>Postscript 1440x720-dpi color printer</td>
</tr>
<tr>
<td><em>IMGB12</em></td>
<td>Postscript 400-dpi color printer</td>
</tr>
<tr>
<td><em>IMGB13</em></td>
<td>Postscript 800-dpi color printer</td>
</tr>
<tr>
<td><em>IMGB14</em></td>
<td>Postscript 600-dpi color printer with larger no-print border</td>
</tr>
<tr>
<td><em>IMGB15</em></td>
<td>Postscript 300-dpi color printer with larger no-print border</td>
</tr>
</tbody>
</table>

---

**IPDS Datastream**

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><em>IMGC01</em></td>
<td>IPDS 240-dpi printer</td>
</tr>
<tr>
<td><em>IMGC02</em></td>
<td>IPDS 300-dpi printer</td>
</tr>
<tr>
<td><em>IMGC03</em></td>
<td>IPDS 600-dpi printer</td>
</tr>
<tr>
<td><em>IMGC04</em></td>
<td>IPDS 1200-dpi printer</td>
</tr>
<tr>
<td><em>IMGC05</em></td>
<td>IPDS 240-dpi printer with no-print border</td>
</tr>
<tr>
<td><em>IMGC06</em></td>
<td>IPDS 300-dpi printer with no-print border</td>
</tr>
<tr>
<td><em>IMGC07</em></td>
<td>IPDS 600-dpi printer with no-print border</td>
</tr>
<tr>
<td><em>IMGC08</em></td>
<td>IPDS 1200-dpi printer with no-print border</td>
</tr>
<tr>
<td><em>IMGC09</em></td>
<td>IPDS 240-dpi printer (IM/1 image only)</td>
</tr>
<tr>
<td><em>IMGC10</em></td>
<td>IPDS 240-dpi printer with no-print border</td>
</tr>
</tbody>
</table>

---

106 IBM Systems - iSeries: i5/OS Commands Starting with CRTMSGF (Create Message File)
Recommended Image Configuration Objects by Printer Table

<table>
<thead>
<tr>
<th>Printer Model</th>
<th>Configuration</th>
</tr>
</thead>
<tbody>
<tr>
<td>Compaq Pagemarc 20</td>
<td>*IMGD01</td>
</tr>
<tr>
<td>Epson EPCL-4 Printer</td>
<td>*IMGA01</td>
</tr>
<tr>
<td>Epson EPCL-5 Printer</td>
<td>*IMGA02</td>
</tr>
<tr>
<td>Epson Stylus Photo with Postscript</td>
<td>*IMGB10</td>
</tr>
<tr>
<td>Epson Stylus Color 600, 800 with Postscript</td>
<td>*IMGB11</td>
</tr>
<tr>
<td>HP Color Laserjet 5</td>
<td>*IMGA04</td>
</tr>
<tr>
<td>HP Color Laserjet 5M</td>
<td>*IMGD04</td>
</tr>
<tr>
<td>HP Deskjet 560C, 820C, 1200C</td>
<td>*IMGA04</td>
</tr>
<tr>
<td>HP Deskjet 500, 600, 1200</td>
<td>*IMGA01</td>
</tr>
<tr>
<td>HP Deskjet 1600C, 1600CN</td>
<td>*IMGA04</td>
</tr>
<tr>
<td>HP Deskjet 1600CM</td>
<td>*IMGD04</td>
</tr>
<tr>
<td>HP Laserjet II, IID, IIP</td>
<td>*IMGA09</td>
</tr>
<tr>
<td>HP Laserjet II, IID, IIP with Postscript</td>
<td>*IMGB01</td>
</tr>
<tr>
<td>HP Laserjet III, IIIID, IIIISi, 4L</td>
<td>*IMGA01</td>
</tr>
<tr>
<td>HP Laserjet III, IIIID, IIIISi, 4L with Postscript</td>
<td>*IMGD01</td>
</tr>
<tr>
<td>HP Laserjet 4, 4P, 4V, 4Si, 4 Plus</td>
<td>*IMGA02</td>
</tr>
<tr>
<td>HP Laserjet 4M, 4MP, 4NV, 4Si MX, 4M Plus</td>
<td>*IMGD02</td>
</tr>
<tr>
<td>HP Laserjet 5, 5P, 5Si</td>
<td>*IMGA02</td>
</tr>
<tr>
<td>HP Laserjet 5M, 5MP, 5Si MX</td>
<td>*IMGD02</td>
</tr>
<tr>
<td>HP Laserjet 6, 6P, 6L</td>
<td>*IMGA02</td>
</tr>
<tr>
<td>HP Laserjet 6M, 6MP</td>
<td>*IMGD02</td>
</tr>
<tr>
<td>IBM 3112, 3116 Page Printer with IPDS feature</td>
<td>*IMGD02</td>
</tr>
<tr>
<td>IBM 3112, 3116 Page Printer (ASCII/LAN)</td>
<td>*IMGA02</td>
</tr>
<tr>
<td>IBM 3112, 3116 Page Printer with Postscript</td>
<td>*IMGD02</td>
</tr>
<tr>
<td>IBM 3130, 3160-1 AF Printer (240-pel mode)</td>
<td>*IMGC01</td>
</tr>
<tr>
<td>IBM 3130 AF Printer (300-pel mode)</td>
<td>*IMGC02</td>
</tr>
<tr>
<td>IBM Infoprint 20 with IPDS feature</td>
<td>*IMGC02</td>
</tr>
<tr>
<td>IBM Infoprint 20 (ASCII)</td>
<td>*IMGA02</td>
</tr>
<tr>
<td>IBM Infoprint 32 with IPDS feature</td>
<td>*IMGC02</td>
</tr>
<tr>
<td>IBM Infoprint 32 (ASCII)</td>
<td>*IMGA02</td>
</tr>
<tr>
<td>IBM Infoprint 60</td>
<td>*IMGC03</td>
</tr>
<tr>
<td>IBM Infoprint 62 Model 2</td>
<td>*IMGC05</td>
</tr>
<tr>
<td>IBM Infoprint 62 Model 3</td>
<td>*IMGC06</td>
</tr>
<tr>
<td>IBM InfoColor 70</td>
<td>*IMGB05</td>
</tr>
<tr>
<td>IBM Infoprint 4000</td>
<td>*IMGC05</td>
</tr>
<tr>
<td>IBM Infoprint 4000 High Resolution</td>
<td>*IMGC06</td>
</tr>
<tr>
<td>IBM 3825, 3827, 3828 AF Printer</td>
<td>*IMGC09</td>
</tr>
<tr>
<td>IBM 3825, 3827, 3828 AF Printer (with AFIG)</td>
<td>*IMGC01</td>
</tr>
<tr>
<td>IBM 3829 AF Printer</td>
<td>*IMGC01</td>
</tr>
<tr>
<td>IBM 3835-001 AF Printer</td>
<td>*IMGC10</td>
</tr>
<tr>
<td>IBM 3835-001 AF Printer (with AFIG)</td>
<td>*IMGC05</td>
</tr>
<tr>
<td>IBM 3835-002, 3900 AF Printer</td>
<td>*IMGC05</td>
</tr>
<tr>
<td>IBM 3912, 3916 Page Printer (ASCII/LAN)</td>
<td>*IMGA01</td>
</tr>
<tr>
<td>IBM 3912, 3916 Page Printer with IPDS feature (twinax)</td>
<td>*IMGC06</td>
</tr>
<tr>
<td>IBM 3930-02 Page Printer (IPDS diskette)</td>
<td>*IMGC01</td>
</tr>
<tr>
<td>IBM 3930-03 Page Printer</td>
<td>*IMGA01</td>
</tr>
<tr>
<td>IBM 3930-03 Page Printer with Postscript</td>
<td>*IMGD01</td>
</tr>
<tr>
<td>IBM 3935 AF Printer</td>
<td>*IMGC02</td>
</tr>
<tr>
<td>IBM 4019 LaserPrinters (HP mode)</td>
<td>*IMGA09</td>
</tr>
</tbody>
</table>
**Internet address (INTNETADR)**

Specifies the internet address of the remote system to which the print request will be sent.

**Note:** This parameter is valid only when RMTSYS(*INTNETADR) and CNNTYPE(*IP) or CNNTYPE(*USRDFN) are specified.

**internet-address**

The internet address is specified in the form `nnn.nnn.nnn.nnn`, where `nnn` is a decimal number ranging from 0 through 255. (An internet address having all binary ones or zeros in the bits of the network or host identifiers portions of the address is not valid.)

Values must be enclosed in apostrophes (') when entered from the command line.

---

**VM/MVS class (CLASS)**

Specifies the VM/MVS SYSOUT class for files sent to a VM/MVS host system.

**Note:** This parameter is valid only when CNNTYPE(*SNA) and DESTTYPE(*S390) are specified.
The class is A.

**character-value**

Specify a distribution class value. Valid values range from A through Z and 0 through 9.

---

**Forms Control Buffer (FCB)**

Specifies the forms control buffer for files sent to a VM/MVS host system.

**Note:** This parameter is valid only when CNNTYPE(*SNA) and DESTTYPE(*S390) are specified.

* **NONE**
  No forms control buffer is used.

* **USRDTA**
  The first 8 characters of the user data (USRDTA) spooled file attribute is the name of the forms control buffer. If the user data is blank, no forms control buffer is used.

* **PRTF**
  The first 8 characters of the printer file used to spool the file is the name of the forms control buffer.

**name**

Specify the name of the forms control buffer to be used.

---

**Destination options (DESTOPT)**

Specifies destination-dependent options. When CNNTYPE(*IP) is specified, the destination-dependent options are added to the control file which is sent to the LPD server. When CNNTYPE(*IP) and DESTTYPE(*NDS) or CNNTYPE(*SNA) is specified, this field is used to determine how spooled files are handled once they are sent to the remote system.

* **NONE**
  No destination options are specified.

* **USRDFNTXT**
  The user-defined text of the user profile when the spooled file was created is used. This value is ignored if CNNTYPE(*IP) and DESTTYPE(*NDS) or CNNTYPE(*SNA) is specified.

* **NOWAIT**
  When CNNTYPE(*IP) and DESTTYPE(*NDS) or CNNTYPE(*SNA) is specified, a value of *NOWAIT indicates that the operating system will no longer keep track of spooled files once they have been sent.

**destination-options**

Specify no more than 128 characters, enclosed in apostrophes.

---

**Print separator page (SEPPAGE)**

Specifies whether or not to request a separator page when printing on a remote system.

* **YES**
  A separator page is requested.

* **NO**
  A separator page is not requested.
User-defined options (USRDFNOPT)

Specifies, for spooled output only, one or more user-defined options to be used by user applications or user-specified programs that process spooled files. A maximum of four user-defined options can be specified.

Single values

*NONE
   No user-defined option is specified.

Other values (up to 4 repetitions)

character-value
   Specify the user-defined option to be used by user applications that process spooled files. All characters are acceptable.

User-defined object (USRDFNOBJ)

Specifies, for spooled output only, the user-defined object to be used by user applications or user-specified programs that process spooled files.

Single values

*NONE
   No user-defined object name is specified.

name
   Specify the name of the user-defined object to be used by user applications or user-specified programs that process spooled files.

Element 1: Object

Qualifier 1: Object

name
   Specify the name of the user-defined object.

Qualifier 2: Library

*LIBL
   All libraries in the library list for the current thread are searched until the first match is found.

*CURLIB
   The current library for the thread is searched. If no library is specified as the current library for the thread, the QGPL library is searched.

name
   Specify the name of the library to be searched.

Element 2: Object type

object-type
   The user object type can be one of the following:
• *DTAARA (Data Area)
• *DTAQ (Data Queue)
• *FILE (File)
• *PSFCFG (PSF Configuration)
• *USRIDX (User Index)
• *USRQ (User Queue)
• *USRSPC (User Space)

**User-defined driver program (USRDRVPGM)**

Specifies the user-defined driver program.

**Note:** This parameter is valid only when RMTSYS is not *NONE.

**Single values**

*NONE  
No user-defined driver program is specified.

**Qualifier 1: User driver program**

*name*  
Specify the name of the user-specified driver program to process spooled files.

**Qualifier 2: Library**

*LIBL  
All libraries in the library list for the current thread are searched until the first match is found.

*CURLIB  
The current library for the thread is searched. If no library is specified as the current library for the thread, the QGPL library is searched.

*name  
Specify the name of the library to be searched.

**Spooled file ASP (SPLFASP)**

Specifies the auxiliary storage pool (ASP) where the spooled files physically reside.

*OUTQASP  
The spooled files reside in the same ASP that the output queue resides in.

*SYSTEM  
The spooled files reside in the system ASP. This value is not allowed if the output queue is in a library on a primary or secondary ASP.

**Text ’description’ (TEXT)**

Specifies the text that briefly describes the object.

*BLANK  
No text is specified.
'description'

Specify no more than 50 characters, enclosed in apostrophes.

---

**Display any file (DSPDTA)**

Specifies whether users who have authority to read the output queue can display the data of any spooled file on the queue or only the data in their own files.

*NO  Users authorized to use the queue can display, copy, or send the data from their own files only, unless they have some special authority.

*YES  Any user having authority to read the queue can display, copy, or send the data of any file on the queue.

*OWNER  The owner of the file or a user with spool control (*SPLCTL) special authority can display, copy, or send the spooled files on the queue.

---

**Job separators (JOBSEP)**

Specifies, for each job with files on the output queue, the number of separators placed at the beginning of the output for the job. Each separator contains information that identifies the job, such as the name of the job, the job user’s name, the job number, and the time and date when the job is run.

This parameter is used only by printer writers, all other types of writers will ignore the value specified for this parameter.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>No job separators are printed before each job’s output.</td>
</tr>
<tr>
<td>*MSG</td>
<td>A message is sent to a message queue notifying the operator of the end of each job. This message queue is identified by the Message queue (MSGQ) parameter of the Start Printer Writer (STRPRTWTR) command.</td>
</tr>
<tr>
<td>0-9</td>
<td>Specify the number of separators to be placed before each job’s output.</td>
</tr>
</tbody>
</table>

---

**Operator controlled (OPRCTL)**

Specifies whether a user who has job control authority is allowed to manage or control the files on this output queue.

*YES  A user with job control authority can control the queue and make changes to the files on the queue.

*NO  This queue and its entries cannot be controlled or changed by users with job control authority unless they also have some other special authority.
Data queue (DTAQ)

Specifies the data queue associated with the output queue. Entries are logged in the data queue when spooled files are in ready (RDY) status on the output queue. A user program can determine when a spooled file is available on an output queue using the Receive Data Queue API (QRCVDTAQ) to receive information from a data queue.

Each time a spooled file on the output queue reaches RDY status, an entry is sent to the data queue. A spooled file can have several changes in status (for example, RDY to held (HLD) to release (RLS) to RDY again) before it is taken off the output queue. These status changes result in entries in the data queue for a spooled file each time the spooled file goes to RDY status.

When the data queue is created using the Create Data Queue (CRTDTAQ) command, the maximum message length (MAXLEN parameter) value should be at least 128 and the sequence (SEQ parameter) value should be *FIFO or *LIFO. More information about data queues on output queues is in the Basic Printing information in the iSeries Information Center at http://www.ibm.com/eserver/iseries/infocenter.

Single values

*NONE No data queue is associated with the output queue.

Qualifier 1: Data queue

name Specify the name of the data queue associated with the output queue.

Qualifier 2: Library

*LIBLE All libraries in the library list for the current thread are searched until the first match is found.

*CURLIB The current library for the thread is searched. If no library is specified as the current library for the thread, the QGPL library is searched.

name Specify the name of the library where the data queue is located.

Authority to check (AUTCHK)

Specifies what type of authorities to the output queue allow the user to control all the files on the queue. Users with some special authority may also be able to control the spooled files.

*OWNER The requester must have ownership authority to the output queue in order to pass the output queue authorization test. The requester can have ownership authority by being the owner of the output queue, sharing a group profile with the queue owner, or running a program that adopts the owner’s authority.

*DTAAUT Any user with add, read, and delete authority to the output queue can control all spooled files on the queue.
Authority (AUT)

Specifies the authority you are giving to users who do not have specific authority for the object, who are not on an authorization list, and whose group profile or supplemental group profiles do not have specific authority for the object.

*USE
Use authority allows the user to perform basic operations on the output queue, such as place spooled files on the queue. *USE authority provides object operational authority, read authority, and execute authority.

*CHANGE
Change authority allows the user to change the output queue description and to control files created by other users if the queue was created with *DTAAUT specified on the Authority to check (AUTCHK) parameter.

*ALL
The user can perform all operations except those limited to the owner or controlled by authorization list management (*AUTLMGT) authority. The user can control the object’s existence, specify the security for the object, change the object, and perform basic functions on the object. The user also can change ownership of the object.

*EXCLUDE
The user cannot access the object.

*LIBCRTAUT
The authority for the object is the same as the value specified on the Create authority (CRTAUT) parameter of the library in which the object is being created. If the value specified on the CRTAUT parameter is changed, the new value will not affect any existing objects.

name
Specify the name of an authorization list to be used for authority to the object. Users included in the authorization list are granted authority to the object as specified in the list. The authorization list must exist when the object is created.

Examples

```
CRTOUTQ OUTQ(DEPTAPRT) AUT(*EXCLUDE) SEQ(*FIFO)
  TEXT('SPECIAL PRINTER FILES FOR DEPTA')
```

This command creates an output queue named DEPTAPRT and puts it in the current library. Because AUT(*EXCLUDE) is specified and OPRCTL(*YES) is assumed, the output queue can be used and controlled only by the user who created the queue and users who have job control authority or spool control authority. Because SEQ(*FIFO) is specified, spooled files are placed in first-in first-out order on the queue. If users in Department A are authorized to use this output queue, the Grant Object Authority (GRTOBJAUT) command must be used to grant them the necessary authority. Data contained in files on this queue can be displayed only by users who own the files, by the owner of the queue, by users with job control authority, or by users with spool control authority. By default, no job separator is printed at the beginning of the output for each job.

Error messages

*ESCAPE Messages

CPF2182
Not authorized to library &1.
CPF2192
Object &1 cannot be created into library &3.

CPF2207
Not authorized to use object &1 in library &3 type *&2.

CPF2212
Not able to allocate library &1.

CPF2402
Library &1 not found

CPF2799
Message queue &1 in library &2 not found.

CPF33F1
Data queue &1 in library &2 not found.

CPF3352
Temporary library &1 invalid for output queue &2.

CPF3353
Output queue &1 in &2 already exists.

CPF3354
Library &1 not found.

CPF3356
Cannot allocate library &1.

CPF3371
Spool user profile QSPL damaged or not found.

CPF34D6
Output queue &1 in &2 not created due to errors.

CPF9818
Object &2 in library &3 not created.
Create Overlay (CRTOVL)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Create Overlay (CRTOVL) command creates an overlay resource from a physical file. The physical file contains the overlay resource information. The overlay resource information, can, for example, come from a S/370 host system and be in the Systems Application Architecture (SAA) format.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>OVL</td>
<td>Overlay</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Overlay</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>FILE</td>
<td>File</td>
<td>Qualified object name</td>
<td>Required, Positional 2</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: File</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>MBR</td>
<td>Member</td>
<td>Name, *OVL</td>
<td>Optional, Positional 3</td>
</tr>
<tr>
<td>DATATYPE</td>
<td>Data type</td>
<td>*AFPDS, *AFPU</td>
<td>Optional</td>
</tr>
<tr>
<td>TEXT</td>
<td>Text ‘description’</td>
<td>Character value, *MBRTXT, *BLANK</td>
<td>Optional</td>
</tr>
<tr>
<td>REPLACE</td>
<td>Replace overlay</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
</tbody>
</table>

Overlay (OVL)

Specifies the overlay that is being created.

This is a required parameter.

Qualifier 1: Overlay

name Specify the name of the overlay.

Qualifier 2: Library

*CURLIB
The current library for the job is used to store the overlay. If no library is specified as the current library for the job, QGPL is used.

name Specify the name of the library where you want to store the overlay.
Source file (FILE)

Specifies the file containing the overlay records sent to this system.

This is a required parameter.

Qualifier 1: File

name Specify the name of the file to be used.

Qualifier 2: Library

*LIBLE All libraries in the library list for the current thread are searched until the first match is found.

*CURLIB

The current library for the job is used to locate the file. If no library is specified as the current library for the job, QGPL is used.

name Specify the library where the file is located.

Source file member (MBR)

Specifies the name of the file member containing the overlay records.

*OVL The name of the file member is specified by the Overlay (OVL) parameter of this command.

name Specify the name of the member in the file specified by the Source file (FILE) parameter.

Data type (DATATYPE)

Specifies the source-type of the input file.

*AFPDS The input is a database file which contains a pre-built Advanced Function Printing Data Stream (AFPDS).

*AFPU The input is a source file created with Advanced Function Printing Utilities for iSeries (AFP Utilities).

Note: This value is valid only if AFP Utilities is installed on your system.

Text 'description' (TEXT)

Specifies the text that briefly describes the object.

*MERTXT The text is taken from the file member being used to create the overlay resource. You can add or change text for a database source member by using the Source Entry Utility (STRSEU) command, or by using either the Add Physical File Member (ADDPFM) command or the Change Physical File Member (CHGPFM) command. If the file is an inline file or a device file, the text is blank.

*BLANK No text is specified.
character-value  
Specify no more than 50 characters of text, enclosed in apostrophes.

Replace overlay (REPLACE)  
Specifies whether an existing overlay with the same name as the one being created is replaced.

*YES  The existing overlay is replaced.
*NO  If an overlay with same name exists on the system, the create operation fails. The existing overlay is not replaced.

Authority (AUT)  
Specifies the authority you are giving to users who do not have specific authority for the object, who are not on an authorization list, and whose group profile or supplemental group profiles do not have specific authority for the object.

*LIBCRTAUT  
The system determines the authority for the object by using the value specified on the Create authority (CRTAUT) parameter on the Create Library command (CRTLIB) for the library containing the object to be created. If the value specified on the Create authority (CRTAUT) parameter is changed, the new value will not affect any existing objects.

*CHANGE  
The user can perform all operations on the object except those limited to the owner or controlled by object existence (*OBJEXIST) and object management (*OBJMGT) authorities. The user can change and perform basic functions on the object. *CHANGE authority provides object operational (*OBJOPR) authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove users.

*ALL  
The user can perform all operations except those limited to the owner or controlled by authorization list management (*AUTLMGT) authority. The user can control the object’s existence, specify the security for the object, change the object, and perform basic functions on the object. The user also can change ownership of the object.

*USE  
The user can perform basic operations on the object, such as running a program or reading a file. The user cannot change the object. Use (*USE) authority provides object operational (*OBJOPR), read (*READ), and execute (*EXECUTE) authorities.

*EXCLUDE  
The user cannot access the object.

name  
Specify the name of an authorization list. Users included on the authorization list are granted authority to the object as specified by the list. The authorization list must exist when the object is created.

Examples

CRTOVL  OVL(MYLIB/MYSIGNTR)  FILE(MYLIB/MYSIGNTR)  MBR(MYSIGNTR)  AUT(*EXCLUDE)  TEXT('representation of my signature')
This command creates the overlay MYSIGNTR into MYLIB. File name MYSIGNTR in library MYLIB with member MYSIGNTR, is used as input. Specifying *EXCLUDE does not allow any other user access to the signature. The text describes the overlay.

**Error messages**

*ESCAPE Messages*

CPF2283
Authorization list &1 does not exist.

CPF8056
File &1 in &2 not a physical file.

CPF88C1
Printer resource type &1 &2 was not created in library &3.

CPF88C2
Data type parameter value of *AFPU incorrect for &1 command.

CPF9809
Library &1 cannot be accessed.

CPF9810
Library &1 not found.

CPF9812
File &1 in library &2 not found.

CPF9822
Not authorized to file &1 in library &2.

CPF9845
Error occurred while opening file &1.

CPF9846
Error while processing file &1 in library &2.

CPF9847
Error occurred while closing file &1 in library &2.

CPF9870
Object &2 type *&5 already exists in library &3.
Create Page Definition (CRTPAGDFN)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Create Page Definition (CRTPAGDFN) command creates a page definition by copying a user-supplied database file to an internal space object. The user must load the source data into the database from a remote system (such as a System/370) or external medium (usually tape) and put it in the SAA format that can be processed by the operating system.

Restrictions: If networking spooled files to a System/370 system, the first two characters of the page definition name must be 'P1'.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>PAGDFN</td>
<td>Page definition</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Page definition</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>FILE</td>
<td>File</td>
<td>Qualified object name</td>
<td>Required, Positional 2</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: File</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>MBR</td>
<td>Member</td>
<td>Name, *PAGDFN</td>
<td>Optional, Positional 3</td>
</tr>
<tr>
<td>TEXT</td>
<td>Text 'description'</td>
<td>Character value, *MBRTXT, *BLANK</td>
<td>Optional</td>
</tr>
<tr>
<td>REPLACE</td>
<td>Replace page definition</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
</tbody>
</table>

Page definition (PAGDFN)

Specifies the page definition to be created.

This is a required parameter.

Qualifier 1: Page definition

name Specify up to eight characters for the name of the page definition.

Qualifier 2: Library

*CURLIB The current library for the job is used to locate the page definition. If no library is specified as the current library for the job, QGPL is used.

name Specify the name of the library where the page definition is located.
File (FILE)
Specifies the data file that contains the page definition records sent to this system.
This is a required parameter.

Qualifier 1: File
name Specify the name of the file to be used.

Qualifier 2: Library
*LIBL All libraries in the library list for the current thread are searched until the first match is found.
*CURLIB The current library for the job is used to locate the data file. If no library is specified as the current library for the job, QGPL is used.
name Specify the name of the library where the data file is located.

Member (MBR)
Specifies the data file member containing the page definition records.

*PAGDFN The name of the data file member is the same as the name specified on the Page definition (PAGDFN) parameter of this command.
name Specify the name of the data file member.

Text ’description’ (TEXT)
Specifies the text that briefly describes the object.

*MBRTXT The text is taken from the data file member used to create the page definition.
*BLANK No text is specified.
character-value Specify no more than 50 characters of text, enclosed in apostrophes.

Replace page definition (REPLACE)
Specifies whether an existing page definition with the same name as the one being created is replaced.

*YES The existing page definition is replaced.
*NO If a page definition with same name exists on the system, the create operation fails. The existing page definition is not replaced.
Authority (AUT)

Specifies the authority you are giving to users who do not have specific authority for the object, who are not on an authorization list, and whose group profile or supplemental group profiles do not have specific authority for the object.

*LIBCRTAUT

The system determines the authority for the object by using the value specified on the Create authority (CRTAUT) parameter on the Create Library command (CRTLIB) for the library containing the object to be created. If the value specified on the Create authority (CRTAUT) parameter is changed, the new value will not affect any existing objects.

*CHANGE

The user can perform all operations on the object except those limited to the owner or controlled by object existence (*OBJEXIST) and object management (*OBJMGT) authorities. The user can change and perform basic functions on the object. *CHANGE authority provides object operational (*OBJOPR) authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove users.

*ALL

The user can perform all operations except those limited to the owner or controlled by authorization list management (*AUTLMGT) authority. The user can control the object’s existence, specify the security for the object, change the object, and perform basic functions on the object. The user also can change ownership of the object.

*USE

The user can perform basic operations on the object, such as running a program or reading a file. The user cannot change the object. Use (*USE) authority provides object operational (*OBJOPR), read (*READ), and execute (*EXECUTE) authorities.

*EXCLUDE

The user cannot access the object.

name

Specify the name of an authorization list. Users included on the authorization list are granted authority to the object as specified by the list. The authorization list must exist when the object is created.

Examples

CRTPAGDFN PAGDFN(*CURLIB/P1DFLT) FILE(*CURLIB/PAGDFNS)
MBR(*PAGDFN) AUT(*EXCLUDE)
TEXT('Default page definition')

This command creates page definition P1DFLT in the current library or in library QGPL if there is no current library. Input is taken from source file PAGDFNS with member P1DFLTF in the current library. Specifying *EXCLUDE for authority restricts use of the object to the owner. The text describes the page definition.

Error messages

*ESCAPE Messages

CPF2283

Authorization list &1 does not exist.
CPF8056
   File &1 in &2 not a physical file.

CPF88C1
   Printer resource type &1 &2 was not created in library &3.

CPF9822
   Not authorized to file &1 in library &2.

CPF9809
   Library &1 cannot be accessed.

CPF9810
   Library &1 not found.

CPF9812
   File &1 in library &2 not found.

CPF9845
   Error occurred while opening file &1.

CPF9846
   Error while processing file &1 in library &2.

CPF9847
   Error occurred while closing file &1 in library &2.

CPF9870
   Object &2 type *&5 already exists in library &3.
Create Page Segment (CRTPAGSEG)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Create Page Segment (CRTPAGSEG) command creates a page segment space object by copying a user-supplied database file to an internal space object. The user must load the page segment resource into the database from a remote system (such as a System/370) or from an external medium (such as a tape) and must put the resource in the SAA format that can be processed by the operating system.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>PAGSEG</td>
<td>Page segment</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Page segment</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>FILE</td>
<td>File</td>
<td>Qualified object name</td>
<td>Required, Positional 2</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: File</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>MBR</td>
<td>Member</td>
<td>Name, *PAGSEG</td>
<td>Optional, Positional 3</td>
</tr>
<tr>
<td>TEXT</td>
<td>Text ‘description’</td>
<td>Character value, *MBRTXT, *BLANK</td>
<td>Optional</td>
</tr>
<tr>
<td>REPLACE</td>
<td>Replace page segment</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
</tbody>
</table>

Page segment (PAGSEG)

Specifies the page segment being created.

This is a required parameter.

Qualifier 1: Page segment

name Specify the name of the page segment.

Qualifier 2: Library

*CURLIB

The current library is used to locate the page segment. If no library is specified as the current library for the job, QGPL is used.

name Specify the library name used to locate the page segment.
Source file (FILE)

Specifies the file containing the page segment records sent to this system.

This is a required parameter.

Qualifier 1: File

name Specify the name of the file to be used.

Qualifier 2: Library

*LIBL All libraries in the library list for the current thread are searched until the first match is found.

*CURLIB The current library for the job is used to locate the file. If no library is specified as the current library for the job, QGPL is used.

name Specify the library where the file is located.

Source file member (MBR)

Specifies the file member containing the page segments records.

*PAGSEG The name of the file member is specified in the Page segment (PAGSEG) parameter.

name Specify the name of the member in the file specified by the Source file (FILE) parameter.

Text ’description’ (TEXT)

Specifies the text that briefly describes the object.

*MBRTXT The text is taken from the file member being used to create the page segment. You can add or change text for a database source member by using the Source Entry Utility (STRSEU) command, or by using either the Add Physical File Member (ADDPFM) command or the Change Physical File Member (CHGPFM) command. If the file is an inline file or a device file, the text is blank.

*BLANK No text is specified.

character-value Specify no more than 50 characters of text, enclosed in apostrophes.

Replace page segment (REPLACE)

Specifies whether an existing page segment with the same name as the one being created is replaced.

*YES The existing page segment is replaced.

*NO If a page segment with same name exists on the system, the create operation fails. The existing page segment is not replaced.
Authority (AUT)

Specifies the authority you are giving to users who do not have specific authority for the object, who are not on an authorization list, and whose group profile or supplemental group profiles do not have specific authority for the object.

*LIBCRTAUT

The system determines the authority for the object by using the value specified on the Create authority (CRTAUT) parameter on the Create Library command (CRTLIB) for the library containing the object to be created. If the value specified on the Create authority (CRTAUT) parameter is changed, the new value will not affect any existing objects.

*CHANGE

The user can perform all operations on the object except those limited to the owner or controlled by object existence (*OBJEXIST) and object management (*OBJMGT) authorities. The user can change and perform basic functions on the object. *CHANGE authority provides object operational (*OBJOPR) authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove users.

*ALL

The user can perform all operations except those limited to the owner or controlled by authorization list management (*AUTLMGT) authority. The user can control the object’s existence, specify the security for the object, change the object, and perform basic functions on the object. The user also can change ownership of the object.

*USE

The user can perform basic operations on the object, such as running a program or reading a file. The user cannot change the object. Use (*USE) authority provides object operational (*OBJOPR), read (*READ), and execute (*EXECUTE) authorities.

*EXCLUDE

The user cannot access the object.

name Specify the name of an authorization list. Users included on the authorization list are granted authority to the object as specified by the list. The authorization list must exist when the object is created.

Examples

CRTPAGSEG PAGSEG(MYLIB/PAGSEG1) FILE(*LIBL/PAGSGMTS) MBR(*PAGSEG) AUT(*ALL) TEXT('canned paragraph 1')

This command creates the page segment PAGSEG1 in MYLIB and uses the PAGSGMTS member, PAGSEG1 as input to the command. Specifying *ALL for the AUT parameter allows any user to perform most object-oriented commands against it. The text contains the description of the object.

Error messages

*ESCAPE Messages

CPF2283

Authorization list &1 does not exist.
CPF8056
File &1 in &2 not a physical file.

CPF88C1
Printer resource type &1 &2 was not created in library &3.

CPF9809
Library &1 cannot be accessed.

CPF9810
Library &1 not found.

CPF9812
File &1 in library &2 not found.

CPF9822
Not authorized to file &1 in library &2.

CPF9845
Error occurred while opening file &1.

CPF9846
Error while processing file &1 in library &2.

CPF9847
Error occurred while closing file &1 in library &2.

CPF9870
Object &2 type *&5 already exists in library &3.
Create Print Descriptor Group (CRTPDG)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Create Print Descriptor Group (CRTPDG) command creates an object of type *PDG into which information about a print descriptor group and its associated print descriptor names can be stored.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>PDG</td>
<td>Print descriptor group</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Print descriptor group</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>TEXT</td>
<td>Text ‘description’</td>
<td>Character value, *BLANK, X’</td>
<td>Optional</td>
</tr>
</tbody>
</table>

Print descriptor group (PDG)

Specifies the name and library of the print descriptor group (PDG) to be created.

`print-descriptor-group-name`

Specify the name of the PDG to be created.

The possible library values are:

*CURLIB

The current library for the job is used to locate the PDG. If no library is specified as the current library for the job, QGPL is used.

library-name

Specify the library name where the created PDG will be stored.

This is a required parameter.

Text ’description’ (TEXT)

Specifies the text that briefly describes the object.

The possible values are:

*BLANK

The text description is left blank.
'description'
   Specify no more than 50 characters of text, enclosed in apostrophes.

Authority (AUT)

Specifies the authority you are giving to users who do not have specific authority for the object, who are not on an authorization list, and whose group profile or supplemental group profiles do not have specific authority for the object.

*LIBCRTAUT

The system determines the authority for the object by using the value specified for the Create authority (CRTAUT) parameter on the Create Library command (CRTLIB) for the library containing the object to be created. If the value specified for the CRTAUT parameter is changed, the new value will not affect any existing objects.

*CHANGE

The user can perform all operations on the object except those limited to the owner or controlled by object existence (*OBJEXIST) and object management (*OBJMGT) authorities. The user can change and perform basic functions on the object. *CHANGE authority provides object operational (*OBJOPR) authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove users.

*ALL

The user can perform all operations except those limited to the owner or controlled by authorization list management (*AUTLMGT) authority. The user can control the object’s existence, specify the security for the object, change the object, and perform basic functions on the object. The user also can change ownership of the object.

*USE

The user can perform basic operations on the object, such as running a program or reading a file. The user cannot change the object. Use (*USE) authority provides object operational (*OBJOPR), read (*READ), and execute (*EXECUTE) authorities.

*EXCLUDE

The user cannot access the object.

name

Specify the name of an authorization list to be used for authority to the object. Users included in the authorization list are granted authority to the object as specified in the list. The authorization list must exist when the object is created.

Examples

CRTPDG   PDG(LETTERS)

This command creates print descriptor group LETTERS.

Error messages

*ESCAPE Messages

CPF2283
   Authorization list &1 does not exist.

CPF6D81
   Print descriptor group &1 not created in library &2.
IBM Systems - iSeries: i5/OS Commands Starting with CRTMSGF (Create Message File)
Create PEX Data (CRTPEXDTA)

Where allowed to run: All environments (*ALL)
Threads: Safe

The Create Performance Data (CRTPEXDTA) command creates the Performance Explorer database files based on the data in a Performance Explorer management collection object (object type *MGTCOL).

Additional information about the Performance Explorer tool can be found in the Performance Management information at http://www.ibm.com/servers/eserver/iseries/perfmgmt/resource.htm.

Restrictions:
1. This command is shipped with public *EXCLUDE authority.
2. The user must have *ADD and *EXECUTE authority to the specified TOLIB library, and *READ and *EXECUTE authority to the FROMLIB library.
3. The user must have *READ authority to the management collection object.
4. To use this command you must have *SERVICE special authority, or be authorized to the Service Trace function of i5/OS through iSeries Navigator’s Application Administration support. The Change Function Usage Information (QSYCHFUI) API, with a function ID of QIBM_SERVICE_TRACE, can also be used to change the list of users that are allowed to perform trace operations.
5. The following user profiles have private authorities to use the command:
   • QPGMR
   • QSRLV

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>FROMMGTCOL</td>
<td>From collection</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: From collection Name</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, QPEXDATA</td>
<td></td>
</tr>
<tr>
<td>TOMBR</td>
<td>To member</td>
<td>Name, *FROMMGTCOL</td>
<td>Optional, Positional 2</td>
</tr>
<tr>
<td>TOLIB</td>
<td>To library</td>
<td>Name, *FROMMGTCOL</td>
<td>Optional, Positional 3</td>
</tr>
<tr>
<td>NBRTHD</td>
<td>Number of threads</td>
<td>1-256, *CALC</td>
<td>Optional</td>
</tr>
<tr>
<td>RPLDTA</td>
<td>Replace data</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
<tr>
<td>TEXT</td>
<td>Text ‘description’</td>
<td>Character value, *BLANK</td>
<td>Optional</td>
</tr>
</tbody>
</table>
From collection (FROMMGTCOL)

Specifies the name of the Performance Explorer management collection object. The data in this object will be stored in the Performance Explorer database files in the specified member.

This is a required parameter.

management_collection_object-name

Specify the name of the management collection object.

The possible library values are:

QPEXDATA

The QPEXDATA library is the recommended library for storing data collected by the Performance Explorer tool. The first time the Performance Explorer tool is used, this library is created for the user.

data-library-name

Specify the name of the library where the management collection object exists.

To member (TOMBR)

Specifies the member name used to store the data in the Performance Explorer database files.

*FROMMGTCOL

The name of the management collection object is used as the member name.

member-name

Specify the name of the member for the database used to store the Performance Explorer data.

To library (TOLIB)

Specifies the library used to store the data in the Performance Explorer database files.

*FROMMGTCOL

The library specified for the management collection object is used.

library-name

Specify the name of the library for the database used to store the Performance Explorer data.

Number of threads (NBRTHD)

Specifies the number of concurrent threads that the CRTPEXDTA command uses to process the data. Specifying a number greater than 1 allows the CRTPEXDTA command to take advantage of available CPU cycles, especially on a multi-processor system. While this may speed up the command processing, it may also degrade the performance of other jobs on the system. You can minimize this impact by changing the priority of the job that runs the CRTPEXDTA command to a higher number. You should also verify the disk subsystem can handle the additional threads. Typically, the CRTPEXDTA command requires one disk arm for each active thread.
*CALC

The system will calculate a reasonable number of threads to do the CRTPExDTA processing which will not use excessive CPU or disk resources. Usually this is one or two threads for each available processor.

number-of-threads

Specify the number of threads for CRTPExDTA to use to process the collected data.

Replace data (RPLDTA)

Specifies whether to replace the data in an existing set of file members with new performance data.

*NO

If a member already exists with the same name, an error message is sent to the user. This prevents the user from inadvertently writing over existing data.

*YES

If a member already exists with the same name, the old data is lost and is replaced by the new data.

Text ’description’ (TEXT)

Specifies the text that briefly describes the type of data collected.

*BLANK

Text is not specified.

’description’

Specify no more than 50 characters of text, enclosed in apostrophes.

Examples

Example 1: Creating PEX Data

CRTPExDTA FROMMGTCOL(QAPEXDTA/MYCOL) TOMBR(TEST)
TOLIB(QAPEXDTA) NBRTHD(2)

This command creates Performance Explorer (PEX) data in member name TEST in library QAPEXDATA. The collected data exists in the management collection object MYCOL found in library QAPEXDTA. Two threads will be used to process the data.

Error messages

None
Create Physical File (CRTPF)

Where allowed to run: All environments (*ALL)
Threadsafe: Conditional

The Create Physical File (CRTPF) command creates a physical file from the information specified on this command and (optionally) from the data description specifications (DDS) contained in a source file.

A physical file is a database file that contains data records. The data records are grouped into physical file members and each member has its own access path to the data. Normally, database files have only one member which, by default, is added to the file when the file is created. If the desired physical file has a record format with only one character field in arrival sequence or if the file is a source file, a DDS source file is not needed. To override attributes of the file after it has been created, use the Override Database File (OVRDBF) command before the file is opened. To change attributes of the file after it has been created, use the Change Physical File (CHGPF) command.

Restrictions:
• This command is conditionally threadsafe. In multithreaded jobs, this command is not threadsafe for distributed files and fails for distributed files that use relational databases of type *SNA.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>FILE</td>
<td>File</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td>Qualifier 1: File</td>
<td>Name</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qualifier 2: Library</td>
<td>Name, *CURLIB</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SRCFILE</td>
<td>Source file</td>
<td>Qualified object name</td>
<td>Optional, Positional 2</td>
</tr>
<tr>
<td>Qualifier 1: Source file</td>
<td>Name, QDDSSRC</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SRCMBR</td>
<td>Source member</td>
<td>Name, *FILE</td>
<td>Optional, Positional 3</td>
</tr>
<tr>
<td>RCDLEN</td>
<td>Record length, if no DDS</td>
<td>Integer</td>
<td>Optional, Positional 4</td>
</tr>
<tr>
<td>GENLVL</td>
<td>Generation severity level</td>
<td>0-30, 20</td>
<td>Optional</td>
</tr>
<tr>
<td>FLAG</td>
<td>Flagging severity level</td>
<td>0-30, 0</td>
<td>Optional</td>
</tr>
<tr>
<td>FILETYPE</td>
<td>File type</td>
<td>*DATA, *SRC</td>
<td>Optional</td>
</tr>
<tr>
<td>MBR</td>
<td>Member, if desired</td>
<td>Name, *FILE, *NONE</td>
<td>Optional</td>
</tr>
<tr>
<td>IGCDTA</td>
<td>User specified DBCS data</td>
<td>*NO, *YES</td>
<td>Optional</td>
</tr>
<tr>
<td>TEXT</td>
<td>Text ‘description’</td>
<td>Character value, *SRCMBRTXT, *BLANK</td>
<td>Optional</td>
</tr>
<tr>
<td>SYSTEM</td>
<td>System</td>
<td>*LCL, *RMT, *FILETYPE</td>
<td>Optional</td>
</tr>
<tr>
<td>EXPDATE</td>
<td>Expiration date for member</td>
<td>Date, *NONE</td>
<td>Optional</td>
</tr>
<tr>
<td>MAXMBRS</td>
<td>Maximum members</td>
<td>Integer, 1, *NOMAX</td>
<td>Optional</td>
</tr>
<tr>
<td>ACCPTHSIZ</td>
<td>Access path size</td>
<td>*MAX1TB, *MAX4GB</td>
<td>Optional</td>
</tr>
</tbody>
</table>

© Copyright IBM Corp. 1998, 2006
<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>PAGE SIZE</td>
<td>Access path logical page size</td>
<td>*KEYLEN, 8, 16, 32, 64, 128, 256, 512</td>
<td>Optional</td>
</tr>
<tr>
<td>MAINT</td>
<td>Access path maintenance</td>
<td>*IMMED, *DLY, *REBLD</td>
<td>Optional</td>
</tr>
<tr>
<td>RECOVER</td>
<td>Access path recovery</td>
<td>*NO, *AFTIPL, *IPL</td>
<td>Optional</td>
</tr>
<tr>
<td>FRC ACC PATH</td>
<td>Force keyed access path</td>
<td>*NO, *YES</td>
<td>Optional</td>
</tr>
</tbody>
</table>
| SIZE           | Member size                        | Single values: *NOMAX  
Other values: Element list | Optional |
|                | Element 1: Initial number of records | 1-2147483646, **1000**                 |        |
|                | Element 2: Increment number of records | Integer, **1000**           |        |
|                | Element 3: Maximum increments      | Integer, **3**                                                        |        |
| ALLOCATE       | Allocate storage                   | *NO, *YES                                                            | Optional |
| CONTIG         | Contiguous storage                 | *NO, *YES                                                            | Optional |
| UNIT           | Preferred storage unit             | 1-255, *ANY                                                          | Optional |
| FRC RATIO      | Records to force a write           | Integer, *NONE                                                       | Optional |
| WAIT FILE      | Maximum file wait time             | Integer, **30**, *IMMED, *CLS                                       | Optional |
| WAIT RCD       | Maximum record wait time           | Integer, **60**, *IMMED, *NOMAX                                       | Optional |
| SHARE          | Share open data path               | *NO, *YES                                                            | Optional |
| DL TPC NT      | Max % deleted records allowed      | 1-100, *NONE                                                         | Optional |
| REUSE DLT      | Reuse deleted records              | *YES, *NO                                                            | Optional |
| SRT SEQ        | Sort sequence                      | Single values: *SRC, *JOB, *LANGID SHR,  
*LANGID UNQ, *HEX  
Other values: Qualified object name | Optional |
|                | Qualifier 1: Sort sequence         | Name                                                                  |        |
|                | Qualifier 2: Library               | Name, *LIBL, *CURLIB                                                  |        |
| LANG ID        | Language ID                        | Character value, *JOB                                                | Optional |
| CCS SID        | Coded character set ID             | Integer, *JOB, *HEX                                                  | Optional |
| AL W UPD       | Allow update operation             | *YES, *NO                                                            | Optional |
| AL W DL T      | Allow delete operation             | *YES, *NO                                                            | Optional |
| LV L CHK       | Record format level check           | *YES, *NO                                                            | Optional |
| NOD GRP        | Node group                         | Single values: *NONE  
Other values: Qualified object name | Optional |
|                | Qualifier 1: Node group            | Name                                                                  |        |
|                | Qualifier 2: Library               | Name, *LIBL, *CURLIB                                                  |        |
| P TN KEY       | Partitioning Key                   | Values (up to 300 repetitions): Name                                 | Optional |

**File (FILE)**

Specifies the physical file to be created.

This is a required parameter.

**Qualifier 1: File**

*name* Specify the name of the physical file to be created.
Qualifier 2: Library

*CURLIB
The current library for the job is where the file is to be located. If no library is specified as the current library for the job, QGPL is used.

name Specify the library where the file is to be located.

Source file (SRCPFLE)

Specifies the source file that contains the data description specifications (DDS) that describe the record format and its fields, and the access path for the file and its members. The specifications that are made in DDS are described in the Database information in the iSeries Information Center at http://www.ibm.com/eserver/iseries/infocenter and the DDS Reference information in the iSeries Information Center at http://www.ibm.com/eserver/iseries/infocenter.

Note: If a value is specified for this parameter, a value cannot be specified for the Record length (RCDLEN) parameter.

Qualifier 1: Source file

QDDSSRC
The DDS source file named QDDSSRC contains the source descriptions used to create the physical file.

name Specify the name of the source file that contains the DDS used to create the physical file.

Qualifier 2: Library

*CURLIB
The current library for the job is where the file is located. If no library is specified as the current library for the job, QGPL is used.

name Specify the library where the file is located.

Source member (SRCMBR)

Specifies the name of the source file member that contains the DDS for the physical file being created; the member is in the source file specified in the SRCFILE parameter (or its default, QDDSSRC). If SRCMBR is specified, RCDLEN cannot be specified.

*FILE The source file member name is the same as that of the physical file specified for the File (FILE) parameter.

name Specify the name of the member in the source file.
Record length (RCDLEN)

Specifies the length (in bytes) of the records stored in the physical file. If RCDLEN and FILETYPE(*DATA) are specified, the physical file is created with a record format that has only one field. The file is then restricted to an arrival sequence access path. The record format and the field are both assigned the same name as that of the file, specified in the FILE parameter. A value ranging from 1 through 32766 bytes can be specified for the record length.

If RCDLEN and FILETYPE(*SRC) are specified, the record format has three fields: source sequence number, date, and source statement. The RCDLEN parameter must provide six positions for the source sequence number, six positions for the date field, and one position for source start, which are required in each record. These fields are defined with fixed attributes and names. If records are copied into the file by the CPYF command and the records are longer than the length specified, the records are truncated on the right.

If RCDLEN is specified, SRCFILE and SRCMBR cannot be specified; RCDLEN is used to specify a fixed record length for the record format when a source file is not needed (when only one field exists in each record or when the file being created is a source file). The high-level language program that processes the file must describe the fields in the record in the program.

Double-Byte Character Set Considerations

If IGCDTA(*NO) is specified, the field is assigned the data type of character whose length is the same as the record length specified. A value ranging from 1 to 32766 bytes can be specified for the record length. If IGCDTA(*YES) is specified, the field is assigned the data type of DBCS-open and a value ranging from 4 to 32766 can be specified.

The RCDLEN parameter must provide six positions for the source sequence number, six positions for the date field, and four positions for source start when FILETYPE(*SRC) and IGCDTA(*YES) are specified.

integer  
Specify the number of bytes in each record.

Generation severity level (GENLVL)

Specifies the severity level at which the create operation fails. If errors occur that have a severity level greater than or equal to this value, the operation ends.

This parameter applies only to messages created while processing DDS source files.

20  
If errors occur in the DDS source file processing with a severity level greater than or equal to 20, the file is not created.

0-30  
Specify the desired severity level value. If 0 is specified, the file is not created. The value specified must be greater than or equal to the value specified for the Flagging severity level (FLAG) parameter.

Flagging severity level (FLAG)

Specifies the minimum severity level of messages to be listed.

0  
All messages are listed.
Specify a number indicating the minimum severity of messages to be listed. The value specified must be greater than or equal to the value specified for the Generation severity level (GENLVL) parameter.

File type (FILETYPE)
Specifies whether each member of the physical file being created contains data records or contains source records (statements) for a program or another file. The file can contain, for example, RPG source statements for an RPG program or DDS source statements for another physical, logical, or device file.

*DATA  The physical file will contain data records.
*SRC   The physical file will contain source records.

Member (MBR)
Specifies the name of the physical file member added when the physical file is created.

*FILE  The name of the member will be the same as the physical file to be created.
*NONE  No physical file member is added when the file is created.
name   Specify the name of the physical file member to be added to the new file.

User specified DBCS data (IGCDTA)
Specifies, for program-described files, whether the file processes double-byte character set (DBCS) data. Specifies, for externally described files, the DBCS attributes of the file.

Note: This parameter has no meaning for physical files created using DDS, because the use of DBCS data is specified in the DDS.

*NO    The file does not contain double-byte character set (DBCS) data.
*YES   The file contains DBCS data.

Double-Byte Character Set Considerations
If the user creates a physical file and specifies the RCDLEN parameter, the system creates a default record format.

- If IGCDTA(*YES) is specified, the default record format can contain DBCS data (as if the record were specified with the DBCS-open (O in column 35 of DDS specification) data type).
- If IGCDTA(*NO) is specified, the default record format cannot contain DBCS data (as if the record were specified with the character (A or blank in column 35 of DDS specification) data type).

The system ignores the IGCDTA parameter value if a value for the RCDLEN parameter is not specified.
The user cannot override the IGCDTA value for a physical file.
Text 'description' (TEXT)

Specifies the text that briefly describes the object.

*SRCMBRTXT
The text is taken from the source file member being used to create the physical file. If the source file is a database file, the text is taken from the source file member. Text can be added or changed for a database source member by using the Source Entry Utility or by using either the Add Physical File Member (ADDPFM) command or the Change Physical File Member (CHGPFM) command. If the source file is an inline file or a device file, the text is blank.

*BLANK
No text is specified.

character-value
Specify no more than 50 characters of text, enclosed in apostrophes.

Source listing options (OPTION)

Specifies the type of output produced when the file is created. A maximum of four of the following values can be specified in any order on this parameter. If neither or both of the values on an option are specified, the first value is used.

Note: The first values on each option are similar to, but are not actually default values, and therefore, cannot be changed with the CHGCMDFT (Change Command Default) command.

Source Listing Option

*SRC or *SOURCE
A printout of the source statements, including a list of errors, is created.

*NOSRC or *NOSOURCE
No printout of the source statements is created unless errors are detected. If errors are detected, they are listed along with the keyword or record format that caused the error.

Program Listing Option

*LIST
An expanded source printout is created, showing a detailed list of the file specifications and the references to other file descriptions.

*NOLIST
The expanded source printout is not created.

Second-Level Message Text Option

*NOSECLVL
The messages section of the data description specifications (DDS) printout does not contain the online help information for messages issued during DDS processing.

*SECLVL
The online help information appears in the DDS printout.

Event File Creation Option
**NOEVENTF**

The compiler does not produce an event file for the CoOperative Development Environment for iSeries (CODE for iSeries)

**EVENTF**

The compiler produces an event file that can be used by the CODE for iSeries product. The event file is created as a member in the file EVFEVENT in your object library. The CODE for iSeries product uses this file to offer error feedback integrated with the CODE for iSeries editor. This value is normally specified by the CODE for iSeries product on your behalf.

---

**System (SYSTEM)**

Specifies whether the physical file is created on the local system or the remote system.

* **LCL** The physical file is created on the local system.
* **RMT** The physical file is created on a remote system. The file specified for the **File (FILE)** parameter must be the name of a distributed data management (DDM) file that identifies the remote system and the name of the physical file being created.

**FILETYPE**

If the file specified for the FILE parameter does not exist on the system, the physical file is created on the local system. Otherwise, the file must be a DDM file, and the physical file is created on a remote system. The DDM file identifies the remote system and the name of the physical file being created.

---

**Expiration date for member (EXPDATE)**

Specifies the expiration date for members of the physical file.

* **NONE** No expiration date is specified.

**date** Specify the date after which the file member cannot be used. The date must be enclosed in apostrophes if date separator characters are used in the value.

---

**Maximum members (MAXMBRS)**

Specifies the maximum number of members that the physical file can contain.

1 Only one member can be contained in the physical file.

* **NOMAX** The number of members that can be contained in the file is the system maximum of 32,767 members.

1-32767 Specify the maximum number of members that can be contained in the physical file.
Access path size (ACCPTHSIZ)

Specifies the maximum size of auxiliary storage that can be occupied by the following kinds of access paths:

- The access paths that are associated with a physical file that has a keyed sequence access path.
- The access paths that are created for referential or unique constraints, and that can be added to this file with the Add Physical File Constraint (ADDPFCST) command.

This parameter does not apply to access paths that are created for logical files or for queries that refer to the data in a physical file.

*MAX1TB
- The access paths associated with this file can occupy a maximum of one terabyte (1,099,511,627,776 bytes) of auxiliary storage.

*MAX4GB
- The access paths associated with this file can occupy a maximum of four gigabytes (4,294,966,272 bytes) of auxiliary storage.

Access path logical page size (PAGESIZE)

Specifies the access path logical page size that is used when the access path is created.

The access path logical page size is used by the system to determine the size of each page of the index. This logical page size is the amount of bytes of the access path that can be moved into the job’s storage pool from the auxiliary storage for a page fault.

*KEYLEN
- The access path logical page size will be determined by the total length of the key, or keys.

8  Logical page size of 8k.
16 Logical page size of 16k.
32 Logical page size of 32k.
64 Logical page size of 64k.
128 Logical page size of 128k.
256 Logical page size of 256k.
512 Logical page size of 512k.

Access path maintenance (MAINT)

Specifies, for files with key fields, the type of access path maintenance used for all members of the physical file.

*IMMED
- The access path is updated each time a record is changed, added, or deleted from a member.

*IMMED must be specified for files that require unique keys.
*REBLD

The access path is completely rebuilt each time a file member is opened. The access path is maintained until the member is closed; then the access path is deleted. *REBLD cannot be specified for files that require unique keys.

*DLY

The maintenance of the access path is delayed until the physical file member is opened for use. Then, the access path is changed only for records that have been added, deleted, or changed since the file was last opened. While the file is open, changes made to its members are immediately reflected in the access paths of those members, no matter what is specified for MAINT. To prevent a lengthy rebuild time when the file is opened, *DLY should be specified only when the number of changes to the access path between successive opens are small; that is, when the file is opened frequently or when the key fields in records for this access path change infrequently.

*REBLD is not valid for access paths that require unique key values.

If the number of changes between a close and the next open reaches approximately 10 percent of the access path size, the system stops saving changes and the access path is completely rebuilt the next time the file is opened.

Access path recovery (RECOVER)

Specifies, for files with immediate or delayed access path maintenance, when recovery processing of the file is performed if the access path is being changed when a system failure occurs. This parameter is valid only for a file with a keyed access path.

If *IMMED is specified for the Access path maintenance (MAINT) parameter, the access path can be rebuilt during initial program load (IPL) (before any user can run a job), or after IPL has ended (during jobs running at the same time), or when the file is next opened. While the access path is being rebuilt, the file cannot be used by any job.

During the IPL, an Override Access Path Recovery display lists those access paths that must be recovered and the RECOVER parameter value for each access path. The user can override the RECOVER parameter value on this display. More information on access paths is in the Backup and Recovery book, SC41-5304.

If *REBLD is specified for the MAINT parameter, the access path is rebuilt the next time its file is opened.

*NO

The access path of the file is rebuilt when the file is opened. *NO is the default for all files that do not require unique keys. The file’s access path, if not valid, is rebuilt when the file is next opened.

Note: *NO is the default for all files that do not require unique keys.

*AFTIPL

The access path of the file is rebuilt after the initial program load (IPL) operation is completed. This option allows other jobs not using this file to start processing immediately after the completion of the IPL. If a job tries to allocate the file while its access path is being rebuilt, a file open exception occurs.

Note: *AFTIPL is the default for all files that require unique keys.

*IPL

The access path of the file is rebuilt during the IPL operation. This ensures that the file’s access path is rebuilt before the first user program tries to use it; however, no jobs can start running until after all files that specify RECOVER(*IPL) have their access paths rebuilt.
**Force keyed access path (FRCACCPTH)**

Specifies, for files with key fields, whether access path changes are forced to auxiliary storage along with the associated records in the file. FRCACCPTH(*YES) minimizes (but does not remove) the possibility that an abnormal job end may cause damage to the access path that requires it to be rebuilt.

**NO** The access path and associated records are not forced to be written to auxiliary storage when the access path is changed.

**YES** The access path and associated records are forced to be written to auxiliary storage when the access path is changed. *YES cannot be specified if *REBLD is specified on the **Access path maintenance (MAINT)** parameter.

FRCACCPTH(*YES) slows the response time of the system if the access path is changed in an interactive job. If the access path is changed frequently, the overall performance of the system is decreased.

---

**Member size (SIZE)**

Specifies the *initial* number of records in each member of the file, the number of records in each part added to the member size, and the number of times the part added is automatically applied. The number of records for each file member is specified as the number of records that can be placed in it (this number includes any deleted records).

When the maximum number of records has been reached, a message (stating that the member is full) is sent to the system operator, giving the choice of ending the request or extending the member size. The operator can extend the member by 10% or by the number of records specified as the increment value, whichever is greater, each time the message is received.

**Single values**

**NOMAX**

The number of records that can be added to each member of the file is not limited by the user. The maximum size of each member is determined by the system. If *NOMAX is specified, *NO must be specified for the **Allocate storage (ALLOCATE)** parameter.

**Element 1: Initial number of records**

Specify the *initial* number of records in each member.

**10000** Initially, up to 10000 records can be written to each member of the file.

**1-2147483646**

Specify the number of records that can be written to each member of the file before the member size is automatically extended.

**Element 2: Increment number of records**

Specify the number of records that are automatically added to the member when the number of records in the member is greater than the initial member size. The minimum size of an increment is 10% of the size of the member at the time the maximum number of records is reached.

**1000** The file size is increased by 10% or 1000 records, whichever is greater.

**integer**

Specify the number of additional records which, if greater than 10% of the size of the member when the maximum number of records is reached, are automatically added to the member.
If the number specified is not greater than 10% of the member size and not equal to zero, the member size is increased by 10%.

If 0 is the specified increment value, the member is not automatically extended. This value must be 0 if the value for the number of increments is 0.

**Element 3: Maximum increments**

Specify the maximum number of increments that can be automatically added to the member.

3

A maximum of 3 increments is automatically added to the member size.

*integer*

Specify the maximum number of increments automatically added to the member size. Valid values range from 0 through 32767. If 0 is specified, the member is not automatically extended.

**Allocate storage (ALLOCATE)**

Specifies whether initial storage space is allocated to each physical file member added to the file. The allocation provides enough space to hold the number of records specified for the Member size (SIZE) parameter. Allocations that occur when a record cannot be added to a member without exceeding its capacity are determined by the system and by the SIZE parameter values.

*NO* The system determines the amount of storage space to allocate to each member added to the file.

*YES* The amount of storage space specified in the first value of the SIZE parameter is allocated each time a new member is added. If *YES* is specified, *NOMAX* must not be specified for the SIZE parameter.

**Contiguous storage (CONTIG)**

Specifies, for each physical file member added to the file, whether all of the records in the initial storage allocation must be stored next to each other.

*NO* Storage of the records next to each other is not required.

*YES* The space containing the records allows the records to be stored next to each other. If the records must be separated, the member is added and a message is sent to the user indicating that storage of the records next to each other is not available.

**Preferred storage unit (UNIT)**

This parameter is no longer supported. It exists solely for compatibility with releases earlier than Version 3 Release 6 Modification 0 of the OS/400 operating system. For information on using auxiliary storage pools (ASPs), refer to the Backup and Recovery book, SC41-5304.

You can specify the value *ANY* or a value ranging from 1 through 255 on this parameter.
Records to force a write (FRCRATIO)

Specifies the number of inserted or updated records that are processed before the records are forced into auxiliary storage. If this physical file is being journalized, either a large number or *NONE should be used. *NONE may cause long synchronization of the journal and physical files.

*NONE
There is no specified force ratio. The system determines when the records are written to auxiliary storage.

integer
Specify the number of inserted or updated records that are processed before the records are written to auxiliary storage.

Maximum file wait time (WAITFILE)

Specifies the number of seconds that the program waits for the file resources to be allocated when the file is opened, or the device or session resources to be allocated when an acquire operation is performed to the file. If the file resources cannot be allocated in the specified wait time, an error message is sent to the program.

*IMMED
The program does not wait. Immediate allocation of file resources is required.

*CLS
The job default wait time is used as the wait time for the file resources to be allocated.

1-32767
Specify the number of seconds to wait for file resources to be allocated.

Maximum record wait time (WAITRCD)

Specifies the number of seconds that the program waits for a record being changed or deleted. If the record cannot be allocated within the specified wait time, an error message is sent to the program.

60
The program waits for 60 seconds for a record being changed or deleted.

*IMMED
The program does not wait; when a record is locked, an immediate allocation of the record is required.

*NOMAX
The wait time is the maximum allowed by the system (32,767 seconds).

1-32767
Specify the number of seconds that the program waits for a record being changed or deleted.

Share open data path (SHARE)

Specifies whether the open data path (ODP) is shared with other programs in the same routing step. When an ODP is shared, the programs accessing the file share facilities such as the file status and the buffer.

Note: This parameter cannot be specified when *NONE is specified for the Member (MBR) parameter.
*NO  The ODP is not shared with other programs in the routing step. A new ODP for the file is created and used every time a program opens the file.

A new ODP for the file is created and used every time a program opens the file with *NO specified for this parameter.

*YES  The same ODP is shared with each program in the job that also specifies *YES when it opens the file.

Max % deleted records allowed (DLTPCT)

Specifies the maximum allowed percentage of deleted records for each member in the physical file. The percentage check is made when the member is closed. If the percentage of deleted records is greater than the value specified on this parameter, a message is sent to the system history log (QHST) to inform the user.

*NONE  The percentage of deleted records in the file members is not checked.

1-100  Specify the largest allowed percentage of deleted records for any member in the file.

Reuse deleted records (REUSEDLT)

Specifies whether the space used by deleted data entries should be reclaimed by future insert requests.

Notes:
• If *YES is specified on this parameter, the key ordering attribute for the physical file in the Data Description Specifications (DDS) source cannot be "FIFO" or "LIFO".
• If a *YES value is specified for this parameter, the arrival order becomes meaningless for a file that reuses deleted record space. Records might not be added at the end of the file.

*NO  The file does not reclaim space used by deleted data entries.

*YES  The file reclaims space used by deleted data entries.

Sort sequence (SRTSEQ)

Specifies the sort sequence used for this file. The sort sequence is used with the LANGID and CCSID parameters to determine which sort sequence table is used.

Single values

*SRC  The table specified in the data description specification (DDS) on the ALTSEQ keyword is used. If ALTSEQ is not used in the DDS, use the value specified for *JOB on this parameter.

*JOB  The sort sequence value used is the value for the job issuing this command to create the physical file.

*LANGIDSHR  The sort sequence table can contain the same weight for multiple characters, and is the shared weighted table associated with the language specified in the LANGID parameter.
**LANGIDUNQ**

The sort sequence table must contain a unique weight for each character in the code page.

**HEX**  A sort sequence table is not used, and the hexadecimal values of the characters are used to determine the sort sequence.

Qualifier 1: Sort sequence

- **name**  Specify the name of the sort sequence table.

Qualifier 2: Library

- **LIBL**  All libraries in the library list for the current thread are searched until the first match is found.

- **CURLIB**  The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

- **name**  Specify the name of the library to be searched.

---

**Language ID (LANGID)**

Specifies the language identifier used when *LANGIDSHR or *LANGIDUNQ is specified on the SRTSEQ parameter. The language identifier is used with the SRTSEQ and CCSID parameters to determine which sort sequence table the file will use.

- **JOB**  The language identifier specified for the job is used.

  - **character-value**  Specify a language identifier. To see a complete list of identifiers when prompting this command, position the cursor on the field for this parameter and press F4 (Prompt).

---

**Coded character set ID (CCSID)**

Specifies the coded character set identifier (CCSID) being used to describe character data in the fields of the file.

This parameter is applicable only if no value is specified for the Source file (SRCFILE) parameter and if a value is specified for the Record length (RCDLEN) parameter. If you specify a value other than the default value (*JOB) on this parameter, the SRCFILE parameter is not used, and a value must be specified for the RCDLEN parameter.

**Note:** A file created with no DDS when FILETYPE(*DATA) is specified has a CCSID of 65535, regardless of the job CCSID value.

- **JOB**  The current job’s default CCSID is used.

- **HEX**  The CCSID 65535 is used, which indicates that character data in the fields is treated as bit data and is not converted.

- **integer**  Specify the CCSID to be used. More information about CCSIDs is in the Globalization topic in the iSeries Information Center at http://www.ibm.com/eserver/iseries/infocenter.
Allow update operation (ALWUPD)

Specifies whether records in this physical file can be updated.

*YES  Records in this physical file can be updated.
*NO   Records in this physical file, or in any logical file that accesses the records in this physical file, cannot be updated.

Allow delete operation (ALWDLT)

Specifies whether records in this physical file can be deleted. Records in a logical file can be deleted only when the records in each physical file on which the logical file is based can be deleted.

*YES  Records in this physical file can be deleted.
*NO   Records in this physical file, or in any logical file that accesses the records in this physical file, cannot be deleted.

Record format level check (LVLCHK)

Specifies whether the level identifiers of the record formats in the physical file are checked when the file is opened by a program.

*YES  The level identifiers of the record formats are checked. If the level identifiers do not all match, an open error message is sent to the program requesting the open operation.
*NO   The level identifiers are not checked when the file is opened.

Node group (NODGRP)

Specifies a node group across which the file is to be distributed.

Single values

*NONE  The file is not a distributed file. All data associated with the file is on the local system.

Qualifier 1: Node group

name  Specify the name of a node group associated with this file.

Qualifier 2: Library

*LIBL  All libraries in the library list for the current thread are searched until the first match is found.
*CURLIB  The current library for the thread is searched. If no library is specified as the current library for the thread, the QGPL library is searched.

name  Specify the name of the library to be searched.
Partitioning Key (PTNKEY)

Specifies the field, or set of fields, that is used as the partition key for distributing data. Up to 300 field names can be specified.

Note: This parameter is not valid if *NONE is specified for the Node group (NODGRP) parameter. If a node group name is specified for the NODGRP parameter, one or more field names must be specified.

name Specify the name of a field to be included in the partition key.

Authority (AUT)

Specifies the authority you are giving to users who do not have specific authority for the object, who are not on an authorization list, and whose group profile or supplemental group profiles do not have specific authority for the object.

*LIBCRTAUT

The system determines the authority for the object by using the value specified for the Create authority (CRTAULT) parameter on the Create Library command (CRTLIB) for the library containing the object to be created. If the value specified for the CRTAULT parameter is changed, the new value will not affect any existing objects.

*CHANGE

The user can perform all operations on the object except those limited to the owner or controlled by object existence (*OBJEXIST) and object management (*OBJMGT) authorities. The user can change and perform basic functions on the object. *CHANGE authority provides object operational (*OBJOPR) authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove users.

*ALL

The user can perform all operations except those limited to the owner or controlled by authorization list management (*AUTLMGT) authority. The user can control the object’s existence, specify the security for the object, change the object, and perform basic functions on the object. The user also can change ownership of the object.

*USE

The user can perform basic operations on the object, such as running a program or reading a file. The user cannot change the object. Use (*USE) authority provides object operational (*OBJOPR), read (*READ), and execute (*EXECUTE) authorities.

*EXCLUDE

The user cannot access the object.

name Specify the name of an authorization list to be used for authority to the object. Users included in the authorization list are granted authority to the object as specified in the list. The authorization list must exist when the object is created.

Examples

Example 1: Creating a Physical File

CRTPF FILE(PAYLIB/PAYTXS) SRCFILE(SRCLIB/PAYTXS) MBR(*NONE) MAXMBRS(5)
This command creates a physical file named PAYTXS in the PAYLIB library. The source descriptions in
the member PAYTXS in source file PAYTXS in the SRCLIB library are used to create the physical file. The
file is created without members (*NONE was specified); therefore, no data can be put into the file until a
member is added later. As many as five members can be contained in the file.

By default, each file member added later will contain data records. The access path of each member is
continuously maintained. Each member can have up to 10,000 records before automatic extensions (three
increments maximum) occur that add 1000 records to the capacity of the member. Storage space for each
member is allocated only as needed, with no restrictions on whether the space is contiguous; there is no
initial storage allocation. The public has object operational, read, add, delete, and update authority for the
file.

Example 2: Creating a Physical File and Member

CRTPF  FILE(ORDERCTL/ORDERS) SRCFILE(ORDERCTL/ORDERSRC)
       SRCMBR(MFGORD)  MAXMBRS(50)  SIZE(1000 100 5)
       ALLOCATE(*YES)

This command creates a physical file and physical file member, both named ORDERS in the ORDERCTL
library. The file and its member are created from the MFGORD source member of the ORDERSRC source
file in the same library. Storage space for the records placed in the file need not be contiguous. Up to 50
members can be contained in the file. The initial allocation of storage provides for up to 1000 records,
and up to five increments of additional space for 100 records each can be added automatically. These
allocation values also apply to each member of this physical file that is added later.

Error messages

*ESCAPE Messages

CPF3204
   Cannot find object needed for file &1 in &2.

CPF323C
   QRECOVERY library could not be allocated.

CPF5702
   File either not DDM file or not found.

CPF7302
   File &1 not created in library &2.
Create Performance Data (CRTPFRDTA)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Create Performance Data (CRTPFRDTA) command creates a set of performance database files from performance information stored in a management collection (*MGTCOL) object. For more information about the database files, see the Performance topic in the iSeries Information Center at the Web site: http://www.ibm.com/eserver/iseries/infocenter.

Performance database files and file members will be created as needed based on the data contained in the management collection object and the information requested on this command. If database files already exist and the requested member exists in any of them, the member will be cleared before the collection is generated.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>FROMMGTCOL</td>
<td>From collection</td>
<td>Single values: *ACTIVE</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Other values: Qualified object name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: From collection</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, QPFRDATA</td>
<td></td>
</tr>
<tr>
<td>TOMBR</td>
<td>To member</td>
<td>Name, *FROMMGTCOL</td>
<td>Optional</td>
</tr>
<tr>
<td>TOLIB</td>
<td>To library</td>
<td>Name, *FROMMGTCOL</td>
<td>Optional</td>
</tr>
<tr>
<td>INTERVAL</td>
<td>Time interval (in minutes)</td>
<td>*FROMMGTCOL, 0.25, 0.5, 1.0, 5.0, 15.0, 30.0, 60.0</td>
<td>Optional</td>
</tr>
<tr>
<td>FROMTIME</td>
<td>Starting date and time</td>
<td>Single values: *FROMMGTCOL Other values: Element list</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Element 1: Starting date</td>
<td>Date</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Starting time</td>
<td>Time</td>
<td></td>
</tr>
<tr>
<td>TOTIME</td>
<td>Ending date and time</td>
<td>Single values: *FROMMGTCOL, *ACTIVE Other values: Element list</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Element 1: Ending date</td>
<td>Date</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Ending time</td>
<td>Time</td>
<td></td>
</tr>
</tbody>
</table>
From collection (FROMMGTCOL)

Specifies the management collection from which a set of performance database files is to be created.

Single values

*ACTIVE
   The currently active collection object will be used.

Qualifier 1: From collection

name Specify the name of the management collection object that is to be used.

Qualifier 2: Library

QPFRDATA
   IBM-supplied performance data library QPFRDATA is to be used to locate the management collection.

name Specify the name of the library for the management collection.

To member (TOMBR)

Specifies the database file member to which the output data is to be written. If a member by this name does not exist in each performance database file, one will be created with the specified name.

*FROMMGTCOL
   The name of the management collection object is used as the performance database file member name.

name Specify the name of the member to which the output should be written.

To library (TOLIB)

Specifies the library where the database files for performance data are to exist. Each file that is not found in the specified library is automatically created in that library.

*FROMMGTCOL
   The performance database files are located or created in the same library as the management collection object (FROMMGTCOL parameter).

name Specify the name of the library where the performance database files are located or should be created.

Text ’description’ (TEXT)

Specifies the text to be used for each member across the set of performance database files associated with the collection.

*SAME
   The value does not change.
FROMMGTCOL
Text associated with the management collection object is used as the member text.

GEN
The following text will be generated - "Created from <collection name> in library <library name>". If the member already exists, no change is made.

BLANK
No text is specified.

character-value
Specify no more that 50 characters of text, enclosed in apostrophes.

Categories to process (CGY)
Specifies the categories in the management collection object which will be processed into database files.

Single values

FROMMGTCOL
All of the categories present in the management collection object will be processed into database files.

Other values (up to 25 repetitions)

category-name
Specify the name of the category of performance information to be processed from the management collection object. Multiple category names may be specified. Valid category names are:

<table>
<thead>
<tr>
<th>Name</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>*APPN</td>
<td>APPN</td>
</tr>
<tr>
<td>*CMNBASE</td>
<td>Communications (Base)</td>
</tr>
<tr>
<td>*CMNSAP</td>
<td>Communications (SAP)</td>
</tr>
<tr>
<td>*CMNSTN</td>
<td>Communications (Station)</td>
</tr>
<tr>
<td>*DISK</td>
<td>Disk</td>
</tr>
<tr>
<td>*DOMINO</td>
<td>Domino for iSeries</td>
</tr>
<tr>
<td>*DPS</td>
<td>Data port services</td>
</tr>
<tr>
<td>*HDWCFG</td>
<td>Hardware</td>
</tr>
<tr>
<td>*HTTP</td>
<td>HTTP Server (powered by Apache)</td>
</tr>
<tr>
<td>*IOPBASE</td>
<td>Input/output processors (Base)</td>
</tr>
<tr>
<td>*IPCS</td>
<td>Network server</td>
</tr>
<tr>
<td>*JOBMI</td>
<td>Jobs (MI tasks and threads)</td>
</tr>
<tr>
<td>*JOBOS</td>
<td>Jobs (operating system)</td>
</tr>
<tr>
<td>*LCLRSP</td>
<td>Local response time</td>
</tr>
<tr>
<td>*LPAR</td>
<td>Logical partition</td>
</tr>
<tr>
<td>*POOL</td>
<td>Pool</td>
</tr>
<tr>
<td>*POOLTUNE</td>
<td>Pool tuning</td>
</tr>
<tr>
<td>*SNA</td>
<td>SNA</td>
</tr>
<tr>
<td>*SNADS</td>
<td>SNADS</td>
</tr>
<tr>
<td>*SUBSYSTEM</td>
<td>Subsystem</td>
</tr>
<tr>
<td>*SYSBUS</td>
<td>System Bus</td>
</tr>
<tr>
<td>*SYSCPU</td>
<td>System CPU</td>
</tr>
<tr>
<td>*SYSLVL</td>
<td>System-level data</td>
</tr>
<tr>
<td>*TCPBASE</td>
<td>TCP/IP (base)</td>
</tr>
<tr>
<td>*TCPIFC</td>
<td>TCP/IP (interface)</td>
</tr>
<tr>
<td>*USRRTNS</td>
<td>User-defined transaction data</td>
</tr>
</tbody>
</table>
**Time interval (in minutes) (INTERVAL)**

Specifies the time interval (in minutes) between successive entries in the database file(s). Within the database, these collection intervals will be identified by interval number and interval time.

Interval numbers will begin with 1 and increment with each interval. Interval time will be based on time at the end of the interval synchronized to the clock time (e.g. if INTERVAL(15) is specified, intervals could be generated as 01:00:00, 01:15:00, 01:30:00, and 01:45:00).

*FROMMGTCOL

The default interval from the management collection object will be used.

*number-of-minutes*

Specify an interval value ranging from 0.25 (15 seconds) through 60 minutes.

---

**Starting date and time (FROMTIME)**

Specifies the starting date and time of the performance data in the management collection object which will be used to create the performance database file(s). This time combined with the interval value will determine the date and time for each database interval.

**Single values**

*FROMMGTCOL

The starting date and time is the date and time that the management collection object was created.

Element 1: Starting date

*date*

Specify the starting date for which collection data is generated. The date must be entered in the format specified by the system values QDATFMT and, if separators are used, QDATSEP.

Element 2: Starting time

*time*

Specify the starting time on the specified starting date for generating the database intervals.

If the starting date is specified and the starting time is not, the starting time will default as follows:

- If the starting date specifies the first date of the collection, the starting time will be set to the start time of the collection.
- If the starting date does not specify the first date of the collection, the starting time will be set to midnight (00:00:00).

---

**Ending date and time (TOTIME)**

Specifies the ending date and time of the last performance data in the management collection object which will be used to create the performance database file(s).

**Single values**

*FROMMGTCOL*

The date and time of the end of the collection in the management collection object will be the ending date and time for data base generation.
**ACTIVE**

The generation of the data base will continue until the currently active collection ends.

This option causes the data base to be generated concurrently with the active collection. The data base will be generated based on the start time specified for any data currently in the management collection object. Additional data will be processed as it is added to the collection object. This will continue until the current collection ends.

Because this option can result in processing for a very long time, it is recommended that TOTIME(*ACTIVE) be specified only when running CRTPFRDTA in a batch job.

**Element 1: Ending date**

date  Specify the ending date for which data from the collection object is used to generate the data base files. The date must be entered in the format specified by the system values QDATFMT and, if separators are used, QDATSEP.

**Element 2: Ending time**

time  Specify the time for the specified ending date for which data from the collection object is used to generate the data base files.

See the description of the Starting time element of the Starting date and time (FROMTIME) parameter for details about how time values can be specified.

If the ending date is specified and the ending time is not, the ending time will default as follows:

- If the ending date specifies the last date of the collection, the ending time will be set to the ending time of the collection.
- If the ending date does not specify the last date of the collection, the ending time will be set to 23:59:59.

---

**Examples**

**Example 1: Generating All Data**

CRTPFRDTA  FROMMGTCOL(Q099365001)  TOMBR(JAN1)  TOLIB(MYLIB)

In this example, the database is generated for all categories contained within the management collection object Q099365001 in library QPFRTDATA. The performance database files will be created into library MYLIB and the collection member name will be JAN1.

Data is generated from the start of data collection within this management collection object to the end of that collection. The database interval is the default collection interval that was specified at the time the collection was started.

**Example 2: Selecting Specific Data**

CRTPFRDTA  FROMMGTCOL(Q099364002)  TOMBR(JAN1J)  TOLIB(MYLIB)
  CGY(*JOBMI)  INTERVAL(15)
  FROMTIME(('01/01/98' '14:00:00'))
  TOTIME(('01/01/98' '16:00:00'))

In this example, only the database file QAPMJOBMI is generated using *JOBMI category information from management collection Q099364002 in library QPFRTDATA. The database interval will be 15 minutes even if the data was collected more frequently (for example, the management collection object may contain data collected every 5 minutes). The generated file will contain only data that was collected between 2:00 PM and 4:00 PM even though the collection object may contain data for a larger time interval.
### Error messages

**ESCAPE Messages**

CPF0A0E
- CRTPFRDTA ended. No files changed.

CPF0A1A
- No active collection.

CPF0A2B
- Not able to process management collection object &1 in library &2.

CPF9801
- Object &2 in library &3 not found.

CPF9810
- Library &1 not found.
Create Program (CRTPGM)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Create Program (CRTPGM) command creates a bound program from a set of modules and binding directories.

Restrictions:
• You must have read (*READ) and add (*ADD) authorities for the library where the program is to be created.
• You must have use (*USE) authority to the specified modules, service programs, and binding directories.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>PGM</td>
<td>Program</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Program</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>MODULE</td>
<td>Module</td>
<td>Single values: *PGM Other values (up to 300 repetitions): Qualified object name</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Module</td>
<td>Generic name, name, *ALL</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB, *USRLIBL</td>
<td></td>
</tr>
<tr>
<td>TEXT</td>
<td>Text ‘description’</td>
<td>Character value, *ENTMODTXT, *BLANK</td>
<td>Optional</td>
</tr>
<tr>
<td>ENTMOD</td>
<td>Program entry procedure module</td>
<td>Single values: *FIRST, *ONLY, *PGM Other values: Qualified object name</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Program entry procedure module</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB, *USRLIBL</td>
<td></td>
</tr>
<tr>
<td>BNDSRVPGM</td>
<td>Bind service program</td>
<td>Single values: *NONE Other values (up to 300 repetitions): Qualified object name</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Bind service program</td>
<td>Generic name, name, *ALL</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL</td>
<td></td>
</tr>
<tr>
<td>BNDDIR</td>
<td>Binding directory</td>
<td>Single values: *NONE Other values (up to 300 repetitions): Qualified object name</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Binding directory</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB, *USRLIBL</td>
<td></td>
</tr>
<tr>
<td>ACTGRP</td>
<td>Activation group</td>
<td>Name, *ENTMOD, *NEW, *CALLER</td>
<td>Optional</td>
</tr>
<tr>
<td>DETAIL</td>
<td>Listing detail</td>
<td>*NONE, *BASIC, *EXTENDED, *FULL</td>
<td>Optional</td>
</tr>
<tr>
<td>ALWUPD</td>
<td>Allow update</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
</tbody>
</table>
### Program (PGM)

Specifies the program object to be created.

This is a required parameter.

**Qualifier 1: Program**

*name* Specify the name of the program to be created.

**Qualifier 2: Library**

* CURLIB

The program object is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

*name* Specify the name of the library where the program object is created.

### Module (MODULE)

Specifies the list of modules that are copied and bound together to create the program object. If duplicate module and library specifications are found, only the first instance of the duplicate module and library is used. Modules in this list are copied into the final program object. Up to 300 names can be specified.

**Single values**

* PGM The name specified for the Program (PGM) parameter is used as the module object name.

**Qualifier 1: Module**

* ALL Find all module objects in the specified library or libraries.

*generic-name* Specify all module objects starting with the characters preceding the * in the specified library or libraries.

*name* Specify the name of the module that is copied to create the program object.

**Qualifier 2: Library**
*LIBL   All libraries in the library list for the current thread are searched until the first match is found.
*CURLIB  The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.
*USRLIBL  Only the libraries in the user portion of the job’s library list are searched.

name   Specify the name of the library to be searched.

Text ’description’ (TEXT)

Specifies text that briefly describes the program object.

*ENTMODTXT  The text description of the module specified for the Program entry procedure module (ENTMOD) parameter is used.

*BLANK   Text is not specified.

character-value   Specify no more than 50 characters of text, enclosed in apostrophes.

Program entry procedure module (ENTMOD)

Specifies the module name that contains the program entry procedure specification to be used for this program.

Single values

*FIRST  The first module found, from the list of modules, that has a program entry procedure specification is selected as the program entry procedure.

*ONLY  Only one module, from the list of modules, can have a specification as the program entry procedure. An error is issued if more than one module is found to have a program entry procedure specification.

*PGM   The name and library specified on the Program (PGM) parameter will be the name and library of the module which has the program entry procedure specification.

Qualifier 1: Program entry procedure module

name   Specify the name of the module containing the program entry procedure specification. If this module is not in the list of modules to be included in this program, it is added to the list of modules. If this module does not have a program entry procedure specification, the program is not created.

Qualifier 2: Library

*LIBL   All libraries in the library list for the current thread are searched until the first match is found.
**CURLIB**

The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

**USRLIBL**

Only the libraries in the user portion of the job’s library list are searched.

name Specify the name of the library to be searched.

---

**Bind service program (BNDSRVPGM)**

Specifies the list of service program exports to examine at bind time to ensure they satisfy any module import requests. The service program exports are checked only if there are unresolved module import requests not satisfied by the set of module exports. Any service program specified on the BNDSRVPGM parameter that satisfies a module import request will be bound to the program being created. The service program name and the library specified on the BNDSRVPGM parameter are saved to be used at run time. Up to 300 names can be specified.

**Single values**

**NONE**

No service program is specified.

**Qualifier 1: Bind service program**

**ALL** Find all service program objects in the specified library or libraries.

Note: This value should only be specified in a user-controlled environment when you know exactly what is getting bound to your program. Specifying *LIBL with *ALL may give you unpredictable results at program run time. Specify the generic service program name or specific libraries to better control what gets bound to your program.

**generic-name**

Specify all service program objects starting with the characters preceding the * in the specified library or libraries.

name Specify the name of the service program to be examined during symbol resolution.

**Qualifier 2: Library**

**LIBL** All libraries in the library list for the current thread are searched until the first match is found.

name Specify the name of the library to be searched.

Note: QTEMP is not a valid library name for this parameter.

---

**Binding directory (BNDDIR)**

Specifies the list of binding directories that are used in symbol resolution. The exports of the modules and service programs in the binding directory are only checked if there are unresolved module import requests that the exports from the modules and service programs (specified in the MODULE or BNDSRVPGM parameters) could not satisfy. Up to 300 names can be specified.

**Single values**
*NONE
   No binding directory is specified.

Qualifier 1: Binding directory
name    Specify the name of the binding directory used in symbol resolution.

Qualifier 2: Library
*LIBL   All libraries in the library list for the current thread are searched until the first match is found.
*CURLIB The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.
*USRLIBL Only the libraries in the user portion of the job’s library list are searched.
name    Specify the name of the library to be searched.

Activation group (ACTGRP)
Specifies the activation group this program is associated with when it is called. An activation group provides:
• Run-time data structures to support the running of programs
• Addressing protection
• A logical boundary for message creation
• A logical boundary for application cleanup processing
*ENTMOD When ACTGRP(*ENTMOD) is specified, the program entry procedure module (ENTMOD parameter) is examined. If the module attribute is RPGLE, CBLLE, or CLE, then ACTGRP(QILE) or ACTGRP(QILETS) is used. QILE is used when STGMDL(*SNGLVL) is specified, and QILETS is used when STGMDL(*TERASPACE) is specified. If the module attribute is not RPGLE, CBLLE, or CLE, then ACTGRP(*NEW) is used.
*NEW    When this program gets called, a new activation group is created. This called program is then associated with the newly created activation group.
*CALLER When this program gets called, the program is activated into the caller’s activation group.
name    Specify the name of the activation group to be used when this program is called.

Creation options (OPTION)
Specifies options to be used when the program object is created.
You can specify up to 5 values for this parameter.

Program Objects
*GEN    A program object is generated.
*NOGEN
A program object is not generated.

Duplicate Procedure Names
*NODUPPROC
During the symbol resolution phase of the binding process, each procedure name that is exported from the modules and programs must be unique.

*DUPPROC
During the symbol resolution phase of the binding process, the procedure names that are exported from the modules and service programs do not have to be unique. When multiple duplicate procedures are allowed, the first exported procedure in the list of specified modules and service programs that matches the import request is the procedure that is selected.

Duplicate Variable Names
*NODUPVAR
During the symbol resolution phase of the binding process, each variable name that is exported from the modules and service programs must be unique.

*DUPVAR
During the symbol resolution phase of the binding process, the variable names that are exported from the modules and service programs do not have to be unique. When multiple duplicate variables are allowed, the first exported variable in the list of specified modules and service programs that matches the import request is the variable that is selected.

Issuing Diagnostic Messages
*WARN
If duplicate variables or procedures are found, a diagnostic message is issued indicating what duplicates were found.

*NOWARN
If duplicate variables or procedures are found, diagnostic messages are not issued.

Resolving References (Imports)
*RSLVREF
All imports must be resolved to exports for the program to be created.

*UNRSLVREF
All imports do not need to resolve to exports for the program to be created. If the program tries to use one of these unresolved imports at run time, a MCH4439 run-time exception is issued.

Listing detail (DETAIL)
Specifies the level of detail to be printed.

*NONE
A listing is not generated.

*BASIC
Contains a listing of the options passed to CRTPGM, and processing statistics. This listing also contains the Brief Summary Table.

*EXTENDED
In addition to the information provided in the *BASIC listing, this listing contains the Extended Summary Table and the Binding Information Listing.
This listing contains the *EXTENDED listing and the Cross-Reference Listing.

Note: If a printed listing is requested, the printer file *LIBL/QSYSPRT is used to generate the listing.

Allow update (ALWUPD)

Specifies whether to allow an update, using the Update Program (UPDPGM) command, of the program being created.

*YES The program can be updated using the UPDPGM command.

*NO The UPDPGM command cannot be used to update the program being created.

Allow *SRVPGM library update (ALWLIBUPD)

Specifies whether to allow the bound service program library name of the program being created to be changed when updated using the Update Program (UPDPGM) command.

*NO The UPDPGM command is not allowed to update the bound service program library names of the program being created, even if *YES is specified for the Allow update (ALWUPD) parameter.

*YES The UPDPGM command is allowed to update the bound service program library names of the program being created when ALWUPD(*YES) is specified.

User profile (USRPRF)

Specifies whether the authority checking done while this program is running includes only the user who is running the program (*USER) or both the user running the program and the program owner (*OWNER). The profiles of the program user or both the program user and the program owner are used to control which objects can be used by the program, including the authority the program has for each object.

*USER The program runs under the user profile of the program’s user.

*OWNER The user profiles of both the program owner and the program user are used when the program is run.

Replace program (REPLACE)

Specifies whether the existing program is replaced if a program by the same name already exists in the specified library.

*YES Replace the existing program by moving it to the QRPLOBJ library. Current activations of the program will continue running, using the version of the program in the QRPLOBJ library.

Note: Both programs must be owned by the same user for the replace to work.
*NO  No replacement occurs. An error message is issued if a program already exists with the name and library specified for the Program (PGM) parameter.

**Authority (AUT)**

Specifies the authority you are giving to users who do not have specific authority for the object, who are not on an authorization list, and whose group profile or supplemental group profiles do not have specific authority for the object.

**LIBCRTAUT**
The system determines the authority for the object by using the value specified for the Create authority (CRTAUT) parameter on the Create Library command (CRTLIB) for the library containing the object to be created. If the value specified for the CRTAUT parameter is changed, the new value will not affect any existing objects.

**CHANGE**
The user can perform all operations on the object except those limited to the owner or controlled by object existence (*OBJEXIST) and object management (*OBJMGT) authorities. The user can change and perform basic functions on the object. *CHANGE authority provides object operational (*OBJOPR) authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove users.

**ALL**
The user can perform all operations except those limited to the owner or controlled by authorization list management (*AUTLMGT) authority. The user can control the object’s existence, specify the security for the object, change the object, and perform basic functions on the object. The user also can change ownership of the object.

**USE**
The user can perform basic operations on the object, such as running a program or reading a file. The user cannot change the object. Use (*USE) authority provides object operational (*OBJOPR), read (*READ), and execute (*EXECUTE) authorities.

**EXCLUDE**
The user cannot access the object.

name  Specify the name of an authorization list to be used for authority to the object. Users included in the authorization list are granted authority to the object as specified in the list. The authorization list must exist when the object is created.

**Target release (TGTRLS)**

Specifies the release of the operating system on which you intend to use the object being created.

When specifying the target-release value, the format VxRxMx is used to specify the release, where Vx is the version, Rx is the release, and Mx is the modification level. For example, V5R3M0 is version 5, release 3, modification 0.

Valid values depend on the current version, release, and modification level of the operating system, and they change with each new release. You can press F4 while prompting this command parameter to see a list of valid target release values.

**CURRENT**
The object is to be used on the release of the operating system currently running on your system. The object can also be used on a system with any subsequent release of the operating system installed.
*PRV  The object is to be used on the previous release with modification level 0 of the operating system. The object can also be used on a system with any subsequent release of the operating system installed.

**character-value**

Specify the release in the format VxRxMx. The object can be used on a system with the specified release or with any subsequent release of the operating system installed.

---

**Allow reinitialization (ALWRINZ)**

Specifies if the static storage of the program is allowed to be reinitialized while it is still active.

*NO  The static storage of the program can not be reinitialized while it is still active.

*YES  The static storage of the program is allowed to be reinitialized while the program is still active.

---

**Storage model (STGMDL)**

Specifies the storage model attribute of the program.

*SNGLVL

The program is created with single-level storage model. When a single-level storage model program is activated and run, it is supplied single-level storage for automatic and static storage. A single-level storage program runs only in a single-level storage activation group.

*TERASPACE

The program is created with teraspace storage model. When a teraspace storage model program is activated and run, it is supplied teraspace storage for automatic and static storage. A teraspace storage program runs only in a teraspace storage activation group.

---

**Interprocedural analysis (IPA)**

Specifies whether interprocedural analysis (IPA) is to be used during the program creation. For more information on IPA, refer to the ILE Concepts book, SC41-5606.

*NO  Interprocedural analysis will not be performed.

*YES  Interprocedural analysis will be performed.

---

**IPA control file (IPACTLFILE)**

Gives the path name of a file which contains interprocedural analysis (IPA) suboption information. This parameter is allowed only when IPA(*YES) is specified.

*NONE  No IPA control file information is to be used when IPA(*YES) is specified.
path-name

Specifies the path name of the IPA control file to use when IPA(*YES) is specified. If the name is qualified it must be enclosed in apostrophes. An example of a qualified IPA control file name is ’/directory1/directory2/myipactlfname’

Examples

CRTPGM   PGM(STAR)

This command creates a program object named STAR in the current library for the job, or library QGPL if there is no current library. The program will be created from one module object that is also named STAR and is located using the current library for the job.

Error messages

*ESCAPE Messages

CPF223E

Authority check for use adopted authority attribute failed.

CPF3C50

Program &1 not created.

CPF5D12

Error encountered during program or service program preparation.
Create Panel Group (CRTPNLGRP)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Create Panel Group (CRTPNLGRP) command creates panel groups that contain online help information, which can be shown in conjunction with your data description specifications (DDS) displays, CL commands, or a search index.

Restrictions:
- You must have read (*READ) and add (*ADD) authorities for the library where the panel group is to be created.
- If the panel group already exists, you must have object existence (*OBJEXIST), object management (*OBJMGT) and read (*READ) authorities for the panel group.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>PNLGRP</td>
<td>Panel group</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Panel group</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>SRCFILE</td>
<td>Source file</td>
<td>Qualified object name</td>
<td>Optional, Positional 2</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Source file</td>
<td>Name, QPNLSRC</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>SRCMBR</td>
<td>Source member</td>
<td>Name, *PNLGRP</td>
<td>Optional, Positional 3</td>
</tr>
<tr>
<td>TEXT</td>
<td>Text ‘description’</td>
<td>Character value, *SRCMBRTXT, *BLANK</td>
<td>Optional</td>
</tr>
<tr>
<td>INCFILE</td>
<td>Include file</td>
<td>Single values: *SRCFILE</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Other values: Qualified object name</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Include file</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Other values: Element list</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 1: Graphic character set</td>
<td>Integer</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Code page</td>
<td>Integer</td>
<td></td>
</tr>
<tr>
<td>REPLACE</td>
<td>Replace panel group</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
</tbody>
</table>
Panel group (PNLGRP)

Specifies the panel group to be created.

This is a required parameter.

Qualifier 1: Panel group

name Specify the name of the panel group to be created.

Qualifier 2: Library

*CURLIB The current library for the job is used to locate the panel group. If no library is specified as the current library for the job, QGPL is used.

name Specify the name of the library where the panel group is located.

Source file (SRCFILE)

Specifies the source file that contains the panel group description source statements. Valid source file record length values range from 13 through 92.

Qualifier 1: Source file

QPNL SRC Source file QPNLSRC contains the panel group description source statements.

name Specify the name of the source file that contains the panel group description source statements.

Qualifier 2: Library

*LIBL All libraries in the library list for the current thread are searched until the first match is found.

*CURLIB The current library for the job is used to locate the source file. If no library is specified as the current library for the job, QGPL is used.

name Specify the name of the library where the source file is located.

Source member (SRCMBR)

Specifies the member of the source file that contains the panel group description.

*PNLGRP The member containing the panel group description has the same name as the name specified for the Panel group (PNLGRP) parameter.

name Specify the name of the member that contains the panel group description.

Text 'description' (TEXT)

Specifies text that briefly describes the panel group.
*SRCMBRTXT
The text associated with the specified source file member is used.

*BLANK
No text is used.

character-value
Specify no more than 50 characters of text, enclosed in apostrophes.

Source listing options (OPTION)
Specifies options for the output produced during the compile. Multiple option values can be specified in any order on this parameter. If neither or both of the values in each group are specified, the underlined value will be used.

Note: The underlined values for this parameter are similar to, but not actually default values, and therefore, cannot be changed with the Change Command Default (CHGCMDFT) command.

Source Listing Option

*SRC or *SOURCE
A source listing is produced.

*NOSRC or *NOSOURCE
No source listing is produced unless errors are detected.

Second-Level Message Text Option

*NOSECLVL
Second-level text is not provided with the first-level text when the messages are printed at the end of the listing.

*SECLVL
Second-level text is provided with the first-level text when the messages are printed at the end of the listing.

Event File Creation Option

*NONEVENTF
The compiler does not produce an event file for the CoOperative Development Environment for iSeries (CODE for iSeries).

*EVENTF
The compiler produces an event file that can be used by the CODE for iSeries product. The event file is created as a member in the file EVFEVENT in your object library. The CODE for iSeries product uses this file to offer error feedback integrated with the CODE for iSeries editor. This value is normally specified by the CODE for iSeries product on your behalf.

Include file (INCFILE)
Specifies the source file containing the members to be included.
Note: If the coded character set identifier (CCSID) of the source file is different than the CCSID of the primary source file specified for the Source file (SRCFILE) parameter, the CCSID is changed to the CCSID of the primary source file. The CCSID must be the same for all source members used to create the object.

Single values

*SRCFILE
The include file is the same file as the file specified for the SRCFILE parameter.

Qualifier 1: Include file

name Specify the name of the source file containing the members to be included.

Qualifier 2: Library

*LIBL All libraries in the library list for the current thread are searched until the first match is found.

*CURLIB The current library for the job is used to locate the source file. If no library is specified as the current library for the job, the QGPL library is used.

name Specify the name of the library where the source file is located.

Character identifier (CHRID)

Specifies whether character identifier (graphic character set and code page) of the dialog variables of the panel group is changed when the panel group is displayed.

Single values

*DEVD No change occurs. The character identifier of the dialog variables and the panel group is the same as the character identifier of the device.

*SYSVAL The character identifier of the dialog variables is changed to the default QCHRID system value for the device. The character identifier of the panel group is the same as the character identifier of the device.

*JOBCCSID The character identifier of the dialog variables is changed from the CCSID of the job to the character identifier of the device. The character identifier of panel group is changed from the CCSID of the source file on the SRCFILE parameter to the character identifier of the device.

*CHRIDCTL The system checks the CHRIDCTL job attribute to determine whether to use *JOBCCSID or *DEVD on the CHRID parameter for this panel group.

Element 1: Graphic character set

integer Specify the graphic character set to be used. Valid values range from 1 through 32767.

Element 2: Code page

integer Specify the code page to be used. Valid values range from 1 through 32767.
Note: Conversion may be necessary depending on the character identifier of the workstation or printer.

Authority (AUT)

Specifies the authority you are giving to users who do not have specific authority for the object, who are not on an authorization list, and whose group profile or supplemental group profiles do not have specific authority for the object.

*LIBCRTAUT

The system determines the authority for the object by using the value specified for the Create authority (CRTAUT) parameter on the Create Library command (CRTLIB) for the library containing the object to be created. If the value specified for the CRTAUT parameter is changed, the new value will not affect any existing objects.

*CHANGE

The user can perform all operations on the object except those limited to the owner or controlled by object existence (*OBJEXIST) and object management (*OBJMGT) authorities. The user can change and perform basic functions on the object. *CHANGE authority provides object operational (*OBJOPR) authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove users.

*ALL

The user can perform all operations except those limited to the owner or controlled by authorization list management (*AUTLMGT) authority. The user can control the object’s existence, specify the security for the object, change the object, and perform basic functions on the object. The user also can change ownership of the object.

*USE

The user can perform basic operations on the object, such as running a program or reading a file. The user cannot change the object. Use (*USE) authority provides object operational (*OBJOPR), read (*READ), and execute (*EXECUTE) authorities.

*EXCLUDE

The user cannot access the object.

name

Specify the name of an authorization list to be used for authority to the object. Users included in the authorization list are granted authority to the object as specified in the list. The authorization list must exist when the object is created.

Replace (REPLACE)

Specifies whether an existing panel group of the same name in the specified library is replaced.

Note: The panel group cannot be replaced if it is in use by this job or another job.

*YES The existing panel group is replaced by moving it to the system library QRPLOBJ.

*NO The existing panel group is not replaced. If such a panel group exists, the create operation fails.

Examples

<table>
<thead>
<tr>
<th>CRTPNLGRP</th>
<th>PNLGRP(PAYLIB/PAYROLL)</th>
<th>SRCFILE(QPNLSRC)</th>
<th>OPTION(*SECLVL)</th>
</tr>
</thead>
</table>

Create Panel Group (CRTPNLGRP) 175
This command creates a panel group named PAYROLL in library PAYLIB, uses source file QPNLSRC in the library list, and prints the second-level message text in the listing.

---

**Error messages**

*ESCAPE Messages*

CPF5A02

Panel group &1 not created in library &2.
The Create Printer File (CRTPRTF) command creates a printer device file from the information specified on this command and, optionally, from the data description specifications (DDS) contained in a source file.

A printer device file is used to send records to a printer device. The printer device file identifies the printer device used and the spooling requirements; it does not contain data.

### Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FILE</strong></td>
<td>File</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: File</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *CURLIB</td>
<td></td>
</tr>
<tr>
<td><strong>SRCFILE</strong></td>
<td>Source file</td>
<td>Single values: *NONE Other values: Qualified object name</td>
<td>Optional, Positional 2</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Source file</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
</tr>
<tr>
<td><strong>SRCMBR</strong></td>
<td>Source member</td>
<td>Name, *FILE</td>
<td>Optional, Positional 3</td>
</tr>
<tr>
<td><strong>GENLVL</strong></td>
<td>Generation severity level</td>
<td>0-30, 20</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>FLAG</strong></td>
<td>Flagging severity level</td>
<td>0-30, 20</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>DEV</strong></td>
<td>Device</td>
<td>Element list</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Element 1: Printer</td>
<td>Name, *JOB, *SYSVAL</td>
<td></td>
</tr>
<tr>
<td><strong>IGCDTA</strong></td>
<td>User specified DBCS data</td>
<td>*NO, *YES</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>IGCSEXN</strong></td>
<td>DBCS extension characters</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>TEXT</strong></td>
<td>Text ‘description’</td>
<td>Character value, *SRCMBRTXT, *BLANK</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>PAGESIZE</strong></td>
<td>Page size</td>
<td>Element list</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Element 1: Length—lines per page</td>
<td>0.001-255.0, 66</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Width—positions per line</td>
<td>0.001-378.0, 132</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 3: Measurement method</td>
<td>*ROWCOL, *UOM</td>
<td></td>
</tr>
<tr>
<td><strong>LPI</strong></td>
<td>Lines per inch</td>
<td>6, 6.0, 3.0, 4.0, 7.5, 7.5, 8.0, 9.0, 12.0</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>CPI</strong></td>
<td>Characters per inch</td>
<td>10, 10.0, 5.0, 12.0, 13.3, 13.3, 15.0, 16.7, 16.7, 18.0, 20.0</td>
<td>Optional</td>
</tr>
<tr>
<td>Keyword</td>
<td>Description</td>
<td>Choices</td>
<td>Notes</td>
</tr>
<tr>
<td>-----------</td>
<td>--------------------------------------------------</td>
<td>----------------------------------------------</td>
<td>--------</td>
</tr>
<tr>
<td>FRONTMGN</td>
<td>Front margin</td>
<td>Single values: *DEVD</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Other values: Element list</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 1: Offset down</td>
<td>0.0-57.79</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Offset across</td>
<td>0.0-57.79</td>
<td></td>
</tr>
<tr>
<td>BACKMGN</td>
<td>Back margin</td>
<td>Single values: *FRONTMGN, *DEVD</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Other values: Element list</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 1: Offset down</td>
<td>0.0-57.79</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Offset across</td>
<td>0.0-57.79</td>
<td></td>
</tr>
<tr>
<td>OVRFLW</td>
<td>Overflow line number</td>
<td>1-255, 60</td>
<td>Optional</td>
</tr>
<tr>
<td>FOLD</td>
<td>Fold records</td>
<td>*NO, *YES</td>
<td>Optional</td>
</tr>
<tr>
<td>RPLUNPR1</td>
<td>Unprintable character action</td>
<td>Single values: *NO</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Other values: Element list</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 1: Replace character</td>
<td>*YES</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Replacement character</td>
<td>X’40’-X’FE’, ‘ ’</td>
<td></td>
</tr>
<tr>
<td>ALIGN</td>
<td>Align page</td>
<td>*NO, *YES</td>
<td>Optional</td>
</tr>
<tr>
<td>CTLCHAR</td>
<td>Control character</td>
<td>*NONE, *FCFC, *MACHINE</td>
<td>Optional</td>
</tr>
<tr>
<td>CHLVAL</td>
<td>Channel values</td>
<td>Single values: *NORMAL</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Other values (up to 12 repetitions): Element list</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 1: Channel</td>
<td>1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Line number for channel</td>
<td>Element list</td>
<td></td>
</tr>
<tr>
<td>FIDELITY</td>
<td>Fidelity</td>
<td>*CONTENT, *ABSOLUTE</td>
<td>Optional</td>
</tr>
<tr>
<td>FORMFEED</td>
<td>Form feed</td>
<td>*DEVD, *AUTOCUT, *CONT, *CUT, *CONT2</td>
<td>Optional</td>
</tr>
<tr>
<td>DRAWER</td>
<td>Source drawer</td>
<td>1-255, 1, *E1, *FORMDF</td>
<td>Optional</td>
</tr>
<tr>
<td>OUTBIN</td>
<td>Output drawer</td>
<td>1-65535, *DEVD</td>
<td>Optional</td>
</tr>
<tr>
<td>FONT</td>
<td>Font</td>
<td>Single values: *CPI, *DEVD</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Other values: Element list</td>
<td></td>
</tr>
<tr>
<td>Keyword</td>
<td>Description</td>
<td>Choices</td>
<td>Notes</td>
</tr>
<tr>
<td>----------</td>
<td>--------------------------</td>
<td>-------------------------------------------------------------------------</td>
<td>---------</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Other values: Element list</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 1: Graphic character set</td>
<td>Integer</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Code page</td>
<td>Integer</td>
<td></td>
</tr>
<tr>
<td>DECFMT</td>
<td>Decimal format</td>
<td>*FILE, *JOB</td>
<td>Optional</td>
</tr>
<tr>
<td>FNTCHRSET</td>
<td>Font character set</td>
<td>Single values: *FONT</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Other values: Element list</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 1: Character set</td>
<td>Qualified object name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Character set</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Code page</td>
<td>Qualified object name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Code page</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 3: Point size</td>
<td>0.1-999.9, *NONE</td>
<td></td>
</tr>
<tr>
<td>CDEFNT</td>
<td>Coded font</td>
<td>Single values: *FNTCHRSET</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Other values: Element list</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 1: Coded font</td>
<td>Qualified object name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Coded font</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Point size</td>
<td>0.1-999.9, *NONE</td>
<td></td>
</tr>
<tr>
<td>TBLREFCHR</td>
<td>Table Reference Characters</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
<tr>
<td>PAGDFN</td>
<td>Page definition</td>
<td>Single values: *NONE</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Other values: Qualified object name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Page definition</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>FORMDF</td>
<td>Form definition</td>
<td>Single values: *NONE, *DEVD</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Other values: Qualified object name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Form definition</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>AFPCHARS</td>
<td>AFP Characters</td>
<td>Single values: *NONE</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Other values (up to 4 repetitions): Character value</td>
<td></td>
</tr>
<tr>
<td>PAGRTT</td>
<td>Degree of page rotation</td>
<td>*AUTO, *DEVD, *COR, 0, 90, 180, 270</td>
<td>Optional</td>
</tr>
<tr>
<td>MULTIP</td>
<td>Pages per side</td>
<td>1-4, 1</td>
<td>Optional</td>
</tr>
<tr>
<td>REDUCE</td>
<td>Reduce output</td>
<td>*TEXT, *NONE</td>
<td>Optional</td>
</tr>
<tr>
<td>PRTTXT</td>
<td>Print text</td>
<td>Character value, *JOB, *BLANK, X’</td>
<td>Optional</td>
</tr>
<tr>
<td>JUSTIFY</td>
<td>Hardware justification</td>
<td>0, 50, 100</td>
<td>Optional</td>
</tr>
<tr>
<td>DUPLEX</td>
<td>Print on both sides</td>
<td>*NO, *YES, *TUMBLE, *FORMDF</td>
<td>Optional</td>
</tr>
<tr>
<td>UOM</td>
<td>Unit of measure</td>
<td>*INCH, *CM</td>
<td>Optional</td>
</tr>
<tr>
<td>FRONTOVL</td>
<td>Front side overlay</td>
<td>Single values: *NONE</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Other values: Element list</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 1: Overlay</td>
<td>Qualified object name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Overlay</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Offset down</td>
<td>0.0-57.79, 0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 3: Offset across</td>
<td>0.0-57.79, 0</td>
<td></td>
</tr>
<tr>
<td>Keyword</td>
<td>Description</td>
<td>Choices</td>
<td>Notes</td>
</tr>
<tr>
<td>---------------</td>
<td>------------------------------</td>
<td>------------------------------------------------------------------------</td>
<td>---------</td>
</tr>
<tr>
<td>BACKOVL</td>
<td>Back side overlay</td>
<td>Single values: *FRONTOVL, *NONE&lt;br&gt;Other values: Element list</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Element 1: Overlay</td>
<td>Qualified object name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Overlay</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Offset down</td>
<td>0.0-57.79, 0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 3: Offset across</td>
<td>0.0-57.79, 0</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 4: Constant back</td>
<td>*NOCONSTANT, *CONSTANT</td>
<td></td>
</tr>
<tr>
<td>CVTLINDTA</td>
<td>Convert line data</td>
<td>*NO, *YES</td>
<td>Optional</td>
</tr>
<tr>
<td>IPDSPASTHR</td>
<td>IPDS pass through</td>
<td>*DEVD, *NO, *YES</td>
<td>Optional</td>
</tr>
<tr>
<td>USRRSCLIBL</td>
<td>User resource library list</td>
<td>Single values: *DEVD, *NONE, *JOBLIBL, *CURLIB&lt;br&gt;Other values (up to 4 repetitions): Name</td>
<td>Optional</td>
</tr>
<tr>
<td>EDGESTITCH</td>
<td>Edge stitch</td>
<td>Single values: *NONE&lt;br&gt;Other values: Element list</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Element 2: Reference edge</td>
<td>0.0-57.79, *DEVD</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 3: Number of staples</td>
<td>1-122, *DEVD</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 4: Staple offsets</td>
<td>Single values: *DEVD&lt;br&gt;Other values (up to 122 repetitions): 0.0-57.79</td>
<td></td>
</tr>
<tr>
<td>SADLSTITCH</td>
<td>Saddle stitch</td>
<td>Single values: *NONE&lt;br&gt;Other values: Element list</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Element 1: Reference edge</td>
<td>*TOP, *LEFT, *DEVD</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Number of staples</td>
<td>1-122, *DEVD</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 3: Staple offsets</td>
<td>Single values: *DEVD&lt;br&gt;Other values (up to 122 repetitions): 0.0-57.79</td>
<td></td>
</tr>
<tr>
<td>FNTRSL</td>
<td>Font resolution for formatting</td>
<td>*DEVD, *SEARCH, 240, 300</td>
<td>Optional</td>
</tr>
<tr>
<td>DFRWRIT</td>
<td>Defer write</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
<tr>
<td>SPOOL</td>
<td>Spool the data</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
<tr>
<td>OUTQ</td>
<td>Spooled output queue</td>
<td>Single values: *JOB, *DEV&lt;br&gt;Other values: Qualified object name</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Spooled output queue</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>FORMTYPE</td>
<td>Form type</td>
<td>Character value, *STD</td>
<td>Optional</td>
</tr>
<tr>
<td>COPIES</td>
<td>Copies</td>
<td>1-255, 1</td>
<td>Optional</td>
</tr>
<tr>
<td>EXPDATE</td>
<td>Expiration date for file</td>
<td>Date, *NONE, *DAYS</td>
<td>Optional</td>
</tr>
<tr>
<td>DAYS</td>
<td>Days until file expires</td>
<td>1-366</td>
<td>Optional</td>
</tr>
<tr>
<td>PAGERANGE</td>
<td>Page range to print</td>
<td>Element list</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Element 1: Starting page</td>
<td>Integer, 1, *ENDPAGE</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Ending page</td>
<td>Integer, *END</td>
<td></td>
</tr>
<tr>
<td>MAXRCDS</td>
<td>Max spooled output records</td>
<td>1-999999, 100000, *NOMAX</td>
<td>Optional</td>
</tr>
<tr>
<td>FILESEP</td>
<td>File separators</td>
<td>0-9, 0</td>
<td>Optional</td>
</tr>
<tr>
<td>SCHEDULE</td>
<td>Spooled output schedule</td>
<td>*FILEEND, *IMMED, *JOBEND</td>
<td>Optional</td>
</tr>
<tr>
<td>HOLD</td>
<td>Hold spooled file</td>
<td>*NO, *YES</td>
<td>Optional</td>
</tr>
<tr>
<td>Keyword</td>
<td>Description</td>
<td>Choices</td>
<td>Notes</td>
</tr>
<tr>
<td>-----------------</td>
<td>------------------------------------------</td>
<td>----------------------------------</td>
<td>-------------</td>
</tr>
<tr>
<td>SAVE</td>
<td>Save spooled file</td>
<td>*NO, *YES</td>
<td>Optional</td>
</tr>
<tr>
<td>OUTPUTY</td>
<td>Output priority (on OUTQ)</td>
<td>*JOB, 1, 2, 3, 4, 5, 6, 7, 8, 9</td>
<td>Optional</td>
</tr>
<tr>
<td>USRDTA</td>
<td>User data</td>
<td>Character value, *SOURCE</td>
<td>Optional</td>
</tr>
<tr>
<td>SPLFOWN</td>
<td>Spool file owner</td>
<td>*CURUSRPRF, *JOB, *CURGRPPRF, *JOBGRPPRF</td>
<td>Optional</td>
</tr>
<tr>
<td>USRDFNOPT</td>
<td>User Defined Option</td>
<td>Single values: *NONE</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Other values (up to 4 repetitions): Character value</td>
<td></td>
</tr>
<tr>
<td>USRDFNDTA</td>
<td>User Defined Data</td>
<td>Character value, *NONE</td>
<td>Optional</td>
</tr>
<tr>
<td>USRDFNOBJ</td>
<td>User Defined Object</td>
<td>Single values: *NONE</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Other values: Element list</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Element 1: Object</td>
<td>Qualified object name</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Qualifier 1: Object</td>
<td>Name</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
</tr>
<tr>
<td>IGCHRRTT</td>
<td>DBCS character rotation</td>
<td>*NO, *YES</td>
<td>Optional</td>
</tr>
<tr>
<td>IGCCPI</td>
<td>DBCS characters per inch</td>
<td>*CPI, *CONDENSED, 5, 6, 10</td>
<td>Optional</td>
</tr>
<tr>
<td>IGCSOSI</td>
<td>DBCS SO/SL spacing</td>
<td>*YES, *NO, *RIGHT</td>
<td>Optional</td>
</tr>
<tr>
<td>IGCCDEFNT</td>
<td>DBCS coded font</td>
<td>Single values: *SYSVAL</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Other values: Element list</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Element 1: DBCS coded font</td>
<td>Qualified object name</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Qualifier 1: DBCS coded font</td>
<td>Name</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Element 2: Point size</td>
<td>0.1-999.9, *NONE</td>
</tr>
<tr>
<td>WAITFILE</td>
<td>Maximum file wait time</td>
<td>Integer, *IMMED, *CLS</td>
<td>Optional</td>
</tr>
<tr>
<td>SHARE</td>
<td>Share open data path</td>
<td>*NO, *YES</td>
<td>Optional</td>
</tr>
<tr>
<td>LVLCHK</td>
<td>Record format level check</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
<tr>
<td>REPLACE</td>
<td>Replace file</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
</tbody>
</table>

**File (FILE)**

Specifies the printer device file to be created.

If the file is used in a high-level language program, the file name should be consistent with the naming rules of that language; otherwise, the file must be renamed in the program itself.

This is a required parameter.

**Qualifier 1: File**

*name* Specify the name of the printer device file.

**Qualifier 2: Library**

* CURLIB

The current library for the job is used to locate the printer device file. If no library is specified as the current library for the job, QGPL is used.

*name* Specify the library where the file is located.
Source file (SRCFILE)

Specifies the source file (if one is specified) containing the data description specifications (DDS) source file used to create the printer device file. More information on the specifications that can be made in DDS is in the Basic Printing information in the iSeries Information Center at http://www.ibm.com/eserver/iseries/infocenter and the DDS Reference information in the iSeries Information Center at http://www.ibm.com/eserver/iseries/infocenter.

Single values

*NONE
There is no DDS source file for this printer device file.

Qualifier 1: Source file

name Specify the name of the source file that contains the DDS for this printer file.

Qualifier 2: Library

*LIBL All libraries in the library list for the current thread are searched until the first match is found.
*CURLIB
The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

name Specify the name of the library to be searched.

Source member (SRCMBR)

Specifies the source file member that contains the data description specifications (DDS) source for the printer file being created.

*FILE The source file member name is the same as the name specified for the File (FILE) parameter.

name Specify the name of the member in the source file.

Generation severity level (GENLVL)

Specifies the severity level of data description specifications (DDS) messages that cause file creation to fail. This parameter applies only to messages created while processing DDS source files.

20 If errors occur in the DDS source file processing with a severity level greater than or equal to 20, the file is not created.

0-30 Specify the desired severity level value. If 0 is specified, the file is not created. The value specified must be greater than or equal to the value specified for the Flagging severity level (FLAG) parameter.
Flagging severity level (FLAG)

Specifies the minimum severity level of messages to be listed.

0       All messages are listed.
0-30    Specify a number indicating the minimum severity of messages to be listed. The value specified must be greater than or equal to the value specified for the Generation severity level (GENLVL) parameter.

Device (DEV)

Specifies the name of a printer device description. For nonspooled output, this identifies the printer device used to produce the printed output. For spooled output, the file is placed on the output queue determined by the OUTQ parameter. If OUTQ(*DEV) is used, the file is placed on the output queue with the same name as the device.

*JOB    The printer associated with the job is the printer device.
*SYSVAL The printer device specified by the system value QPRTDEV at the time the job is started is the printer device.
name    Specify the name of the printer device used with the printer file.

Printer device type (DEVTYP)

Specifies the type of data stream created for the printer device file.

*SCS    An SNA character stream (SCS) is created. This parameter must be specified when using the 3287, 3812 SCS, 3816 SCS, 4214, 4234 SCS, 4245, 5219, 5224, 5225, 5256, 5262, or 6262 work station printers.
    • If *SCS is specified and the spooled printer file is directed to an IPDS* printer, the SCS printer file is converted to emulate an IPDS printer file. More information is in the Basic Printing information in the iSeries Information Center at http://www.ibm.com/eserver/iseries/infocenter.

Double-Byte Character Set Consideration

When using the 5553 and 5583 DBCS-capable printers, DEVTYP(*SCS) must be specified. An SNA Character Stream (SCS) data stream is created.

*IPDS   An Intelligent Printer Data Stream (IPDS) is created. This parameter can be specified when using an IPDS printer.

*USERASCII

An ASCII data stream is placed on a spooled output queue. You are responsible for placing the entire hexadecimal data stream in the buffer, since the iSeries system does not change or validate the values that are passed.

*AFPDS  An advanced function print data stream (AFPDS) is created. Some systems refer to this data stream as MODCA-P.
*AFPDSLINE
   Mixed data (line data and AFPDS data) is created. This value can be specified when using any
   printer supported by PSF. The printer must be configured with AFP(*YES).

*LINE
   Line data is created. This value can be specified when using any printer supported by PSF. The
   printer must be configured with AFP(*YES).

User specified DBCS data (IGCDTA)

Specifies, for program-described files, whether the file processes double-byte character set (DBCS) data. Specifies, for externally described files, the DBCS attributes of the file.

For program-described files

*NO   The file does not process double-byte character set (DBCS) data.

*YES  The file processes double-byte character set (DBCS) data.

For externally-described files

*NO   The double-byte character set (DBCS) attributes of the file are specified in the field descriptions.

*YES  DBCS attributes in addition to those specified in the field descriptions include: (1) putting the
       data description specification (DDS) keyword for alternative data type (IGCALTTYP) into effect,
       and (2) identifying DBCS attributes of fields, values, or messages.

DBCS extension characters (IGCEXNCHR)

Specifies whether the system processes double-byte character set (DBCS) extended characters. When processing DBCS extended characters, the device requires the assistance of the system. The system must tell the device what the character looks like before the device can display or print the character. Extended characters are stored in a DBCS font table, not in the DBCS device. Extended character processing is a function of the operating system that is required to make characters stored in a DBCS font table available to a DBCS device.

*YES   The system processes DBCS extended characters.

*NO    The system does not process DBCS extended characters; it prints extended characters as the
       undefined character.

Text 'description' (TEXT)

Specifies the text that briefly describes the object.

*SRCMBRTXT
   If the source file is a database file, the text is taken from the source file member used to create the
   file. If the source file is an inline file or a device file, the text is blank.

*BLANK
   No text is specified.

color-value
   Specify no more than 50 characters of text, enclosed in apostrophes.
Source listing options (OPTION)

Specifies the type of output produced when the file is created. A maximum of four of the following values can be specified in any order on this parameter. If neither or both of the values on an option are specified, the first value listed for the option is used.

**Note:** The first values on each option are similar to, but are not actually default values, and therefore, cannot be changed with the CHGCMDFT (Change Command Default) command.

Source Listing Option

*SRC or *SOURCE
- A printout of the source statements, including a list of errors, is created.

*NOSRC or *NOSOURCE
- No printout of the source statements is created unless errors are detected. If errors are detected, they are listed along with the keyword or record format that caused the error.

Program Listing Option

*LIST
- An expanded source printout is created, showing a detailed list of the file specifications and the references to other file descriptions.

*NOLIST
- The expanded source printout is not created.

Second-Level Message Text Option

*NOSCLVL
- The messages section of the data description specifications (DDS) printout does not contain the online help information for messages issued during DDS processing.

*SECLVL
- The online help information appears in the DDS printout.

Event File Creation Option

*NONEVENTF
- The compiler does not produce an event file for the CoOperative Development Environment for iSeries (CODE for iSeries) product.

*EVENTF
- The compiler produces an event file that can be used by the CODE for iSeries product. The event file is created as a member in the file EVFEVENT in your object library. The CODE for iSeries product uses this file to offer error feedback integrated with the CODE for iSeries editor. This value is normally specified by the CODE for iSeries product on your behalf.

Page size (PAGESIZE)

Specifies the length and width of the printer forms used by this device file. The length is specified in lines per page or by the units specified for the UOM parameter. The width is specified in print positions (characters) per line or by the units specified for the **Unit of measure (UOM)** parameter.
The page size must be specified with reference to the way the data is printed on the page. For example, if using 8.5 inch wide by 11.0 inch long forms and printing at 6 lines per inch with a 10-pitch font, specify PAGESIZE(66 85) PAGRTT(0). However, to rotate the page, specify the page size for an 11.0 inch wide by 8.5 inch long page and enter PAGESIZE(51 110) PAGRTT(90).

**Note:** Specify PAGRTT(*AUTO) or PAGRTT(*DEVD) and PRTQLTY(*DRAFT) on this command to enable automatic reduction or rotation if the data does not fit on the paper.

Specify PAGRTT(*COR) on this command to enable automatic reduction whether or not the data fits on the paper.

**Element 1: Length—lines per page**

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>66</td>
<td>The page length is 66 print lines per page.</td>
</tr>
<tr>
<td>0.001-255.0</td>
<td>Specify the page length that is used by this printer file. The value specified must not exceed the actual length of the forms used.</td>
</tr>
</tbody>
</table>

**Element 2: Width—positions per line**

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>132</td>
<td>The page width is 132 printed characters per line.</td>
</tr>
<tr>
<td>0.001-378.0</td>
<td>Specify the page width that is used by this printer file. The value specified must not exceed the actual width of the forms used.</td>
</tr>
</tbody>
</table>

**Element 3: Measurement method**

<table>
<thead>
<tr>
<th>Measurement method</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>ROWCOL</strong></td>
<td>Page length and page width are measured as numbers of rows and columns.</td>
</tr>
<tr>
<td><strong>UOM</strong></td>
<td>Page length and page width are measured in the units specified for the UOM parameter.</td>
</tr>
</tbody>
</table>

**Lines per inch (LPI)**

Specifies the line spacing setting on the printer, in lines per inch, to be used by this device file.

The line spacing on the 5256 printer must be set manually. When the lines per inch (LPI) value on this parameter changes (from the value on the previous printer file), an inquiry message is sent to the message queue associated with the printer that requests a change to the LPI value.

The line spacing on the 4214, 4224, 4230, 4234, 4245, and 5262 Printers is set by a print command. These also allow setting the lines per inch spacing on the control panel of the printer. The lines per inch value must not be set at the printer. If the LPI value is overridden at the control panel, the system overrides the value set with the LPI value of the next printer file received.

<table>
<thead>
<tr>
<th>Element</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>The line spacing on the printer is 6 lines per inch.</td>
</tr>
<tr>
<td>3</td>
<td>The line spacing on the printer is 3 lines per inch. This value is valid only for double-byte character set (DBCS) printers.</td>
</tr>
<tr>
<td>4</td>
<td>The line spacing on the printer is 4 lines per inch.</td>
</tr>
<tr>
<td>7.5</td>
<td>The line spacing on the printer is 7.5 lines per inch. This value is valid only for double-byte character set (DBCS) printers.</td>
</tr>
<tr>
<td>8</td>
<td>The line spacing on the printer is 8 lines per inch.</td>
</tr>
</tbody>
</table>
Note: When printing double-byte character set (DBCS) data for a file specified with LPI(8), use double spacing. Otherwise, the DBCS data does not print correctly. Alphanumeric data, however, prints correctly in single spacing when LPI(8) is specified.

<table>
<thead>
<tr>
<th>Line spacing</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>9</td>
<td>The line spacing on the printer is 9 lines per inch.</td>
</tr>
<tr>
<td>12</td>
<td>The line spacing on the printer is 12 lines per inch.</td>
</tr>
</tbody>
</table>

**Characters per inch (CPI)**

Specifies the printer character density, in characters per inch.

For the printers that support fonts, the value specified in the font special value implies the CPI. If FONT(*CPI) is specified, the font used is based on the CPI value. The following diagram describes the default font ID for each CPI value:

<table>
<thead>
<tr>
<th>CPI</th>
<th>FONT ID</th>
<th>DEFAULT</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>245</td>
<td>DEFAULT</td>
</tr>
<tr>
<td>10</td>
<td>011</td>
<td></td>
</tr>
<tr>
<td>12</td>
<td>087</td>
<td></td>
</tr>
<tr>
<td>13.3</td>
<td>204</td>
<td></td>
</tr>
<tr>
<td>15</td>
<td>222</td>
<td></td>
</tr>
<tr>
<td>16.7</td>
<td>400</td>
<td></td>
</tr>
<tr>
<td>18</td>
<td>252</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>281</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>CPI</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>10</td>
<td>Character density is 10 characters per inch.</td>
</tr>
<tr>
<td>5</td>
<td>Character density is 5 characters per inch.</td>
</tr>
<tr>
<td>12</td>
<td>Character density is 12 characters per inch.</td>
</tr>
<tr>
<td>13.3</td>
<td>Character density is 13.3 characters per inch. This value is valid only for double-byte character set (DBCS) printers.</td>
</tr>
<tr>
<td>15</td>
<td>Character density is 15 characters per inch.</td>
</tr>
<tr>
<td>16.7</td>
<td>Character density is 16.7 characters per inch.</td>
</tr>
<tr>
<td>18</td>
<td>Character density is 18 characters per inch. This value is valid only for double-byte character set (DBCS) printers.</td>
</tr>
<tr>
<td>20</td>
<td>Character density is 20 characters per inch. This value is valid only for double-byte character set (DBCS) printers.</td>
</tr>
</tbody>
</table>

**Front margin (FRONTMGN)**

Specifies the offset, down and across, of the origin from the edge on the front side of the paper. The offsets are in the units of measure specified on the UOM parameter. This parameter can only be used for printer files with DEVTYPE(*AFPDS) specified.

**Single values**

| *DEVD | The no-print border from the printer is used to place the text on the page when printing to a printer configured with AFP(*YES). A margin of 0 is used for IPDS printers without a no-print border, or which are configured with AFP(*NO). |

**Element 1: Offset down**
0.0-57.79
Specify the offset of the origin from the top of the page. If *CM (centimeter) is specified for the Unit of measure (UOM) parameter, valid values range from 0 through 57.79. If *INCH is specified for the UOM parameter, valid values range from 0 through 22.75.

Element 2: Offset across
0.0-57.79
Specify the offset of the origin from the left side of the page. If *CM (centimeter) is specified for the Unit of measure (UOM) parameter, valid values range from 0 through 57.79. If *INCH is specified for the UOM parameter, valid values range from 0 through 22.75.

Back margin (BACKMGN)
Specifies the offset, down and across, of the origin from the edge on the back side of the paper. The offsets are in the units of measure specified on the UOM parameter. This parameter can only be used for printer files with DEVTYPE(*AFPDS) specified.

Single values
*FRONTMGN
The offsets specified for the Front margin (FRONTMGN) parameter are used.

*DEVD
The no-print border from the printer is used to place the text on the page when printing to a printer configured with AFP(*YES). A margin of 0 is used for IPDS printers without a no-print border, or which are configured with AFP(*NO).

Element 1: Offset down
0.0-57.79
Specify the offset of the origin from the top of the page. If *CM (centimeter) is specified for the Unit of measure (UOM) parameter, valid values range from 0 through 57.79. If *INCH is specified for the UOM parameter, valid values range from 0 through 22.75.

Element 2: Offset across
0.0-57.79
Specify the offset of the origin from the left side of the page. If *CM (centimeter) is specified for the Unit of measure (UOM) parameter, valid values range from 0 through 57.79. If *INCH is specified for the UOM parameter, valid values range from 0 through 22.75.

Overflow line number (OVRFLW)
Specifies the line number on the page at which printer overflow to a new page occurs. Overflow is signaled when the specified line number becomes the current line, whether or not printing has occurred on that line.

60 After line 60 has been reached or printed, the printer overflows to a new page.

1-255 Specify the overflow line number. The value specified must not exceed the page length specified in the Page size (PAGESIZE) parameter. Margins specified for the printer file are ignored when determining overflow.
Fold records (FOLD)

Specifies whether all positions in a record are printed when the record length is greater than the page width. If *IPDS is specified on the Printer device type (DEVTYPE) parameter, this parameter is ignored.

When folding is specified and a record exceeds the page width, any portion of the record that cannot be printed on the first line continues (is folded) on the next line or lines until the entire record has been printed.

The FOLD parameter is ignored under the following conditions:
• When DEVTYPE(*SCS) is not specified.
• When printing through OfficeVision.
• When in the S/36 execution environment.

Double-Byte Character Set Considerations

The system ignores this parameter when printing double-byte character set (DBCS) files. The system assumes that DBCS records fit on a printed line. If the record exceeds the page width, the system continues printing the record on the next line.

*NO  Records are not folded; if a record is longer than the page width, only the first part of the record that fits on one line is printed.

*YES  Records whose length is greater than the page width are folded on the following lines.

Unprintable character action (RPLUNPRT)

Specifies whether unprintable characters are replaced and which substitution character (if any) is used for unprintable characters. An unprintable character is a character that the printer is not able to print.

Double-Byte Character Set Considerations

For double-byte character set (DBCS) data, an unprintable character is one that cannot be processed. When using DBCS-capable printers, consider the following:
• If IGCEXNCHR(*YES) is also specified, the system replaces unprintable extension characters with DBCS underline characters. There may be some cases in which the system is unable to replace an unprintable character with a DBCS underline character. In this case, the undefined character is printed.
• If IGCEXNCHR(*NO) is also specified, the device replaces all extension characters with the undefined character. Choosing a blank as the replacement character for alphanumeric characters might improve system performance.

More information is in the Basic Printing information in the iSeries Information Center at http://www.ibm.com/eserver/iseries/infocenter.

Single values

*NO  Unprintable characters are not replaced. When an unprintable character is detected, a message is sent to the program.

Element 1: Replace character
*YES  Unprintable characters are replaced. The program is not notified when unprintable characters are detected.

Element 2: Replacement character

' ' A blank (X'40') is used as the substitution character when an unprintable character is detected.

*X'40'-'X'FE' Specify the replacement character that is used each time an unprintable character is detected. This character is used only if *YES is also specified in this parameter. Valid values range from 40 through 99 and A1 through FE.

Align page (ALIGN)

Specifies whether each page must be aligned in the printer before printing is started. If ALIGN(*YES) and SPOOL(*NO) are specified, and forms alignment is required, the system sends a message to the message queue specified in the printer device description and waits for a reply to the message. When spool (*YES) is specified on the CRTPRTF command and ALIGN(*FILE) is specified on the STRPRTWTR command, then this parameter is used to determine whether an alignment message is sent by the system.

This parameter is ignored when cut sheets are used (spooled and direct output). Page alignment can be done only for text-only files. Page alignment cannot be done for print jobs containing graphics or bar codes.

*NO  Page alignment is not required.

*YES  Page alignment is required before the output is printed.

Control character (CTLCHAR)

Specifies whether the printer device file supports input with print control characters.

*NONE  No print control characters are passed in the printed data.

*FCFC  The first character of every record contains an American National Standards Institute (ANSI) forms control character. If *FCFC is specified, the record length must include one position for the first-character forms-control code, which is passed in the printed data.

*MACHINE  The first character of every record contains a machine code control character. If *MACHINE is specified, the record length must include one extra position for the first character forms control code. This value is not valid for externally described printer files.

If TBLREFCHR(*YES) is also specified, then the record length must include two extra positions for the control character and the table reference character.
Channel values (CHLVAL)

Specifies a list of channel numbers with their assigned line numbers. Use this parameter only if *FCFC is specified for the Control character (CTLCHAR) parameter.

Single values

*NORMAL  
The default values for skipping to channel identifiers are used. The default values are found in the following table:

<table>
<thead>
<tr>
<th>Code</th>
<th>Action before Printing a Line</th>
</tr>
</thead>
<tbody>
<tr>
<td>' '</td>
<td>Space one line (blank code)</td>
</tr>
<tr>
<td>0</td>
<td>Space two lines</td>
</tr>
<tr>
<td>-</td>
<td>Space three lines</td>
</tr>
<tr>
<td>+</td>
<td>Suppress space</td>
</tr>
<tr>
<td>1</td>
<td>Skip to line 1</td>
</tr>
<tr>
<td>2-11</td>
<td>Space one line</td>
</tr>
<tr>
<td>12</td>
<td>Skip to overflow line (OVRFLW parameter)</td>
</tr>
</tbody>
</table>

Element 1: Channel

1-12  Specify an American National Standard channel number to be associated with a corresponding ‘skip to’ line number. Valid values for this parameter range from 1 through 12, corresponding to channels 1 through 12. The CHLVAL parameter associates the channel number with a page line number. For example, if you specify CHLVAL(2 20), channel identifier 2 is allocated with line number 20; therefore, if you place the forms-control 2 in the first position of a record, the printer skips to line 20 before printing the line.

Note: If the printer stops and the next record processed has a channel value forms-control number that is the same value as the line number the printer is on, the printer advances to that value (line number) on the next page. However, if the printer is positioned at the top of the page (line number one) and the channel value forms-control value is associated with line number one, the printer does not advance to a new a new page.

If no line number is specified for a channel identifier, and that channel identifier is encountered in the data, a default of ‘space one line’ before printing is used. Each channel number can be specified only once.

Element 2: Line number for channel

1-255  Specify the line number assigned for the channel number in the same list. Valid line numbers range from 1 through 255. If no line number is assigned to a channel number, and that channel number is encountered in the data, a default of ‘space one line’ before printing is used.

Fidelity (FIDELITY)

Specifies whether printing continues when print errors are found for printers configured with AFP(*YES).

*CONTENT  
Printing continues when errors are found.

*ABSOLUTE  
Printing stops when errors are found.
Print quality (PRTQLTY)

Specifies, for the 3812 SCS, 3816 SCS, 4214, 4224, 4230, 4234, and 5219 printers, the quality of print produced.

Some non-impact IPDS printers support a toner miser feature. This feature is device dependent. Specifying *DRAFT for the PRTQLTY value will activate the toner miser feature on those printers that support it.

For the 5219 Printer, different print qualities are produced by varying the speed at which the print ribbon advances. Quality mode (*STD or *NLQ) results in normal print ribbon advancement. In draft mode (*DRAFT), the ribbon advances at a rate of one-third the distance it advances in quality mode. The 5219 Printer has a conserve ribbon switch that overrides the value of *DRAFT specified by this parameter.

For the 3812 SCS and 3816 SCS Printers, the automatic hardware selection of computer output reduction printing selected through soft switches on the printers occurs only when *DRAFT is specified for PRTQLTY and PAGRTT is *DEVD. If PAGRTT(*COR) is specified, the PRTQLTY parameter does not affect the printed output.

For the 4224, 4230, and 4234 Printers, standard print quality is produced by varying the density of the dot matrix pattern used to create printable characters. Standard mode (*STD) is the normal mode. Quality mode (*NLQ) requires multiple passes by the printer to produce a line of data. Draft mode (*DRAFT) results in high-speed printing.

For the 4214 printer, only draft (*DRAFT), quality (*NLQ), and device default (*DEVD) modes are supported. Other values are set to quality (*NLQ) mode.

More information about the valid values for the 4214, 4224, 4230, 4234, and 5219 Printers is in the Printer Device Programming Manual.

Notes:
- For the 4214 Printer, quality mode (*STD or *NLQ) is only supported for 10 and 12 characters per inch. If PRTQLTY(*STD or *NLQ) and 5, 15, or 16.7 characters per inch is specified, the data is printed in draft mode.
- For the 4234 Printer, only a limited character set (62 characters) is supported when PRTQLTY(*DRAFT) is specified. A description of the character set supported with draft print quality is in the 4234 Printer Operator’s Guide.
- For the 4224 and 4230 printers, the fonts supported are not available for all three print qualities. The OCR-A and OCR-B fonts are supported only with PRTQLTY(*NLQ). The Courier and Essay fonts are available only with PRTQLTY(*NLQ) and PRTQLTY(*STD). The Gothic font is available only with PRTQLTY(*DRAFT) or PRTQLTY(*FASTDRAFT). If there is a mismatch between the print quality and the font selected, the font is changed to match the print quality.
- For the 4214 printer, only draft (*DRAFT), quality (*NLQ), and device default (*DEVD) modes are supported. Other values are set to quality (*NLQ) mode.

Form feed (FORMFEED)

Specifies the form feed attachment used by this printer device file.

*DEVD

  The forms are fed into the printer as specified in the printer device description.
Continuous forms are used by the printer. The tractor feed attachment must be on the printer device.

Continuous forms are used by the printer. The form is fed from the secondary tractor feed attachment. The secondary tractor feed attachment must be on the printer device.

Single-cut sheets are used by the printer. Each sheet must be manually loaded.

Single-cut sheets are semiautomatically fed into the printer. The sheet-feed attachment must be on the printer device.

Source drawer (DRAWER)

Specifies the source drawer used when cut sheets are fed into the printer (specified by FORMFEED(*AUTOCUT)).

1 The paper is fed from the first drawer on the sheet-feed paper handler.

*E1 The envelopes are fed from the envelope drawer on the sheet feed paper handler.

*FORMDF

The paper is fed from the source drawer specified in the form definition. If a form definition is not specified, then source drawer 1 is used.

1-255 Specify the drawer from which the paper is fed.

Output bin (OUTBIN)

Specifies the destination of the output on printers capable of multiple output bins.

*DEVD

The destination of the output is the device default output bin.

1-65535 Specify the output bin for the destination of the output.

Font identifier (FONT)

 Specifies the font identifier and point size used with this printer device file.

More information about the valid font identifiers, the display value, the characters per inch value implied with each font style, a description of each font style, and whether the font is supported on a particular printer is in the Printer Device Programming Manual.

Note: Some fonts can be substituted by the printer. Consult the various printer reference guides for details.

Single values

*CPI The identifier of the font with the specified pitch (characters per inch (CPI)) is used.
*DEVD
The font identifier and point size specified in the device description are used.

Element 1: Identifier
identifier
Specify the numeric font identifier to be used with this printer device file.

Element 2: Point size
*NONE
No point size is specified; the system selects one based on the type of printer used.

0.1-999.9
Specify a point size.

Character identifier (CHRID)
Specifies the character identifier (graphic character set and code page) for the printer file. This parameter allows you to print text that has different character identifier coding. The value specified for this parameter is used to instruct the printer device to interpret the hexadecimal byte string to print the same characters that were intended when the text was created. More information about the character identifier is in the Basic Printing information in the iSeries Information Center at http://www.ibm.com/eserver/iseries/infocenter.

Single values
*DEVD
The default value from the Character identifier (CHRID) parameter that the device is designed to handle is used. The *DEVD value means character selection is normal because the file has the same character identifier as the printer device default.

*SYSVAL
The value on the Character identifier (CHRID) parameter specified for the system on which the application runs is used.

*JOBCCSID
The character identifier for the printer file is taken from the coded character set identifier (CCSID) of the job.

Note: The *JOBCCSID special value, either specified directly on the CHRID command parameter or on the CHRIDCTL job attribute when the *CHRIDCTL special value is specified for the CHRID command parameter, is not allowed if the file was created on a system at an earlier release level than V2R3M0. A file created prior to V2R3M0 will not be tagged with a CCSID and cannot be used in combination with the *JOBCCSID support.

*CHRIDCTL
The system checks the CHRIDCTL job definition attribute to determine whether to use *JOBCCSID or *DEVD on the CHRID command parameter for this file.

Element 1: Graphic character set
integer
Specify the graphic character set value that matches the printer.

Element 2: Code page
integer

Specify the code page value that matches the printer. Valid values range from 1 through 32767.

---

**Decimal format (DECFMT)**

Specifies which decimal format value is used when editing numeric fields with the EDTCDE DDS keyword. The decimal format value determines the use of commas and periods for the decimal position and three digit positional separators on edited fields.

*JOB Use the decimal format value from the DECFMT job attribute when the file is opened.

*FILE Use the decimal format value stored with the file when the file was created.

---

**Font character set (FNTCHRSET)**

Specifies a downloaded font consisting of a character set and code page. This parameter can only be used for printer files with DEVTYPE(*AFPDS) specified.

**Single values**

*FONT

The value specified for the Font identifier (FONT) parameter is used.

**Element 1: Character set**

Qualifier 1: Character set

*name Specify the name of the font character set.

Qualifier 2: Library

*LIBL All libraries in the library list for the current thread are searched until the first match is found.

*CURLIB The current library for the job is used to locate the font character set. If no library is specified as the current library for the job, the QGPL library is used.

*name Specify the name of the library where the font character set is located.

**Element 2: Code page**

Qualifier 1: Code page

*name Specify the name of the code page.
Qualifier 2: Library

*LIBL  All libraries in the library list for the current thread are searched until the first match is found.

*CURLIB  The current library for the job is used to locate the code page name. If no library is specified as the current library for the job, the QGPL library is used.

name  Specify the name of the library where the code page object is located.

Element 3: Point size

*NONE  The point size is supplied by the system and is determined by the specified font character set.

0.1-999.9  Specify the point size to be used.

Note: The point size parameter is only used when an outlined font is named, in other cases it is ignored.

Coded font (CDEFNT)

Specifies the coded font that the system uses for single-byte character set (SBCS) printing. This parameter can only be used for printer files with DEVTYPE(*AFPDS) specified.

Single values

*FNTCHRSET  The font specified for the Font character set (FNTCHRSET) parameter is used.

Element 1: Coded font

Qualifier 1: Coded font

name  Specify the name of the coded font.

Qualifier 2: Library

*LIBL  All libraries in the library list for the current thread are searched until the first match is found.

*CURLIB  The current library for the job is used to locate the coded font object. If no library is specified as the current library for the job, the QGPL library is used.

name  Specify the name of the library where the coded font object is located.

Element 2: Point size

*NONE  The point size is supplied by the system and is determined by the specified font character set.
Specify the point size to be used.

**Note:** The point size parameter is only used when an outlined font is named, in other cases it is ignored.

---

### Table Reference Characters (TBLREFCHR)

Specifies whether table reference characters are present in the line data.

- **NO**  
  No table reference character is present in line data.

- **YES**  
  Table reference characters are present in line data.

  If forms control characters are used with the data, the table reference character follows the forms control character but precedes the data bytes. If forms control characters are not used, the table reference character is the first byte of the data record. As with forms control character, if table reference characters are used, every data record must contain a TRC byte.

---

### Page definition (PAGDFN)

Specifies the page definition to be used to format line data.

You can specify a page definition with *LINE, *AFPSDLINE, or *USERASCII data. PSF/400 will convert the line data and page definition to IPDS.

When you specify a page definition on the printer file, some printer file parameters will be ignored when the spooled file is printed by PSF/400. The following print file parameters will be ignored:

- CDEFNT
- CHRID
- CPI
- FNTCHRSET
- FOLD
- FONT
- LPI
- MULTIUP
- PAGESIZE
- PAGRTT
- REDUCE

**Single values**

- **NONE**  
  No page definition is specified.

  Because PSF/400 requires a page definition when *LINE or *AFPSDLINE is specified, an inline page definition is built from the print file parameters and passed to PSF/400 when *NONE is specified.

**Qualifier 1: Page definition**
**name**  Specify the name of the page definition that must exist in the library specified. Valid values range from 1 to 8 characters. Device type *AFPDSLNE, *LINE, or *USERASCII must be specified when using a page definition.

**Qualifier 2: Library**

**LIBL**  All libraries in the library list for the current thread are searched until the first match is found.

**CURLIB**  The current library for the thread is searched. If no library is specified as the current library for the thread, the QGPL library is searched.

**name**  Specify the name of the library to be searched.

---

**Form definition (FORMDF)**

Specifies the form definition to use when printing the file. A form definition is a resource object that defines the characteristics of the form, including overlays, position of page data on the form, and number of copies of pages and modifications to pages. The form definition is located inline with the file being printed, or in a library.

When you specify a form definition (*DEVD or form definition name) on the printer file, some printer file parameters will be ignored when the spooled file is printed by PSF/400. The following print file parameters will be ignored:

- **DUPLEX** (If *FORMDF specified)
- **DRAWER** (If *FORMDF specified)
- **PAGRTT**
- **PRTQLTY**
- **FORMFEED**
- **FRONTMGN**
- **BACKMGN**
- **MULTIUP**
- **REDUCE**
- **CORNERSTPL**
- **EDGESTITCH**
- **SADLSTITCH**

**Single values**

**NONE**  No form definition is used.

Because PSF/400 requires a form definition, an inline form definition is built from the print file parameters and passed to PSF/400 when *NONE is specified.

**DEVD**  The name of the form definition is specified in the printer device description.

**Qualifier 1: Form definition**

**name**  Specify the name of the form definition that must exist in the library specified. Valid values range from 1 to 8 characters.

**Qualifier 2: Library**
**LIBL**  All libraries in the library list for the current thread are searched until the first match is found.

**CURLIB**  The current library for the thread is searched. If no library is specified as the current library for the thread, the QGPL library is searched.

*name*  Specify the name of the library to be searched.

---

### AFP Characters (AFPCHARS)

Specifies one or more AFP characters (coded fonts) to be used with line data and a page definition.

**Single values**

**NONE**  No AFP characters (coded fonts) specified.

**Other values (up to 4 repetitions)**

*character-value*

Specify up to four 4-byte names of coded fonts to be specified with the line data and a page definition. The 4-byte names are concatenated to X0 to identify up to four coded fonts which are to be used when TBLREFCHR is being used within the data.

---

### Degree of page rotation (PAGRTT)

Specifies the degree of rotation of the text on the page with respect to the way the page is loaded into the printer. See the note under the PAGESIZE parameter for directions on specifying page size when rotating the page.

Specify *AUTO* or *DEVD* for this parameter and PRTQLTY(*DRAFT) on this command to enable automatic rotation if the data does not fit on the paper.

**AUTO**  Indicates that automatic rotation of output is done to fit the printed data on the form. If rotation does not accomplish this, computer output reduction is performed automatically (regardless of the print quality being used). This parameter is valid only for printers supporting rotation.

**COR**  Computer output reduction (COR) is used. COR allows printed output intended for a 13.2 inch wide by 11.0 inch long page to be printed on an 11.0 inch wide by 8.5 in long 8.5 inch wide by 11.0 inch long page.

For computer output reduction printing, the following operations are done for cut-sheet IPDS printers:

- Automatic rotation to *COR* is not done if the file contains graphics, bar codes, variable LPI, variable font, variable page rotations, or variable drawer.

- The text is rotated 90 degrees clockwise from the 0 degree rotation position (lower left corner of the first edge loaded into the printer).

  **Note:** For landscape paper on non-impact continuous form printers, the rotation is counter-clockwise from the 0 degree rotation position (upper right corner of the first edge loaded into the printer).

- A top and left margin of 0.5 inches is added to the printed output.
• The 12-pitch fonts are changed to a 15-pitch font and 15-pitch fonts are changed to a 20-pitch font. All other font widths are changed to a 13.3-pitch font, except for the 4028 printer where they are changed to a 15-pitch font.
• Vertical spacing (specified by the LPI parameter) is 70 percent of the normal spacing.
• The page size is set to 8.5 inches wide by 11 inches long.

*DEVD
The operating system sends a device default rotation value to the printer. Page rotation is dependent on your printer’s specifications. See your printer or printer emulation documentation to determine how page rotation is affected.

0 No rotation is done.
90 The text is rotated 90 degrees clockwise.
180 The text is rotated 180 degrees clockwise.
270 The text is rotated 270 degrees clockwise.

Pages per side (MULTIUP)
Specifies, for spooled output only, whether or not multiple pages of output are printed on 1 physical page.

Note: Overlays are not reduced when more than one page is printed on a side.

For more information and examples see the Basic Printing information in the iSeries Information Center at http://www.ibm.com/eserver/iseries/infocenter.

1 One page of output is printed on one physical sheet of paper.
2 Two pages of output are printed on 1 physical sheet of paper.
3 Three pages of output are printed on 1 physical sheet of paper.
4 Four pages of output are printed on 1 physical sheet of paper.

Reduce output (REDUCE)
Specifies whether or not to reduce the output when doing multiple up printing.

For more information and examples see the Basic Printing information in the iSeries Information Center at http://www.ibm.com/eserver/iseries/infocenter.

*TEXT
The text output is reduced when doing multiple up printing.

*NONE
The output is not reduced when doing multiple up printing.

Print text (PRTTXT)
Specifies the text that is printed at the bottom of each page of printed output and on separator pages.
*JOB  The value from the current job is used.

*BLANK  No text is printed.

*character-value  Specify no more than 30 characters of text, enclosed in apostrophes.

---

**Hardware justification (JUSTIFY)**

Specifies the printing positions of the characters on a page to control the degree of print alignment at the right-hand margin. Justification is done to the record length on the printer file opened.

**Note:** The JUSTIFY parameter is supported only on the 3812 SCS, 3816 SCS, and 5219 Printers.

- 0  No justification occurs.
- 50  Spaces are added to the blanks in the text so that the right margin is more closely aligned but not flush.
- 100  The text is expanded by spaces (added where the blanks already exist) until the right margin is flush.

---

**Print on both sides (DUPLEX)**

Specifies whether output is printed on one side or two sides of the paper.

- *NO  The output is printed on one side of the paper.
- *YES  The output is printed on both sides of the paper, with the top of each printed page at the same end of the sheet of paper. This is usually done for output that is bound at the side.
- *TUMBLE  The output is printed on both sides of the paper, with the top of one printed page at the opposite end from the top of the other printed page. This is usually done for output that is bound at the top.
- *FORMDF  The output is printed on both sides of the paper if the duplex value is specified in the form definition. If a form definition is not specified, then the output is printed on one side of the paper.

---

**Unit of measure (UOM)**

Specifies the unit of measurement to be used.

- *INCH  The inch is used as the unit of measurement.
- *CM  The centimeter is used as the unit of measurement.
Front side overlay (FRONTOVL)

Specifies the object that contains both the overlay that is printed on the front side of the page and the offset, down and across, from the point of origin used when the overlay is printed.

Single values

*NONE
   No overlay is used.

Element 1: Overlay

Qualifier 1: Overlay

name  Specify the name of the overlay.

Qualifier 2: Library

*LIBL  All libraries in the library list for the current thread are searched until the first match is found.

*CURLIB  The current library for the job is used to locate the overlay. If no library is specified as the current library for the job, the QGPL library is used.

name  Specify the name of the library where the overlay is located.

Element 2: Offset down

0  No offset down from the point of origin is used.

0.0-57.79  Specify the offset down from the point of origin at which to begin printing the overlay. If UOM(*CM) is specified, valid values range from 0 through 57.79, and if UOM(*INCH) is specified, valid values range from 0 through 22.75.

Element 3: Offset across

0  No offset across from the point of origin is used.

0.0-57.79  Specify the offset across from the point of origin at which to begin printing the overlay. If UOM(*CM) is specified, valid values range from 0 through 57.79, and if UOM(*INCH) is specified, valid values range from 0 through 22.75.

Back side overlay (BACKOVL)

Specifies the object that contains both the overlay that is printed on the back side of the page and the offset, down and across, from the point of origin used when the overlay is printed.

The constant back function allows you to print overlays on blank pages without adding blank pages to the print application. Specifying the constant back function would cause, for each page generated by the application program, a blank page to be generated onto which the specified back overlay could be...
The generated blank pages are called constant forms because no variable data from the user’s program is printed on the pages. The constant back function is only supported for duplex printing. It is ignored when DUPEX(*NO) is specified on the printer file.

Note that the offset down and offset across values are ignored when *CONSTANT is specified for constant back. An offset of 0.0 is assumed for these values.

Single values

*FRONTOVL
The values specified for the Front side overlay (FRONTOVL) parameter are used.

*NONE
No overlay is used.

Element 1: Overlay

Qualifier 1: Overlay

name Specify the name of the overlay.

Qualifier 2: Library

*LIBL All libraries in the library list for the current thread are searched until the first match is found.

*CURLIB
The current library for the job is used to locate the overlay. If no library is specified as the current library for the job, the QGPL library is used.

name Specify the name of the library where the overlay is located.

Element 2: Offset down

0 No offset down from the point of origin is used.

0.0-57.79
Specify the offset down from the point of origin at which to begin printing the overlay. If UOM(*CM) is specified, valid values range from 0 through 57.79, and if UOM(*INCH) is specified, valid values range from 0 through 22.75.

Element 3: Offset across

0 No offset across from the point of origin is used.

0.0-57.79
Specify the offset across from the point of origin at which to begin printing the overlay. If UOM(*CM) is specified, valid values range from 0 through 57.79, and if UOM(*INCH) is specified, valid values range from 0 through 22.75.

Element 4: Constant back

*NOCONSTANT
No constant back is specified.

*CONSTANT
Constant back is specified.
Convert line data (CVTLINDTA)

Specifies whether line data and a page definition should be converted to AFPDS before the data is spooled.

*NO  No AFPDS conversion is done.
*YES Specifies that AFPDS conversion is to be done on the line data and page definition before the data is spooled.

IPDS pass through (IPDSPASTHR)

Specifies whether IPDS (intelligent printer data stream) pass-through is done for the spooled file.

*DEVD The value specified for IPDSPASTHR in the PSF configuration object specified for a printer device description is used. If no PSF configuration object is specified for the device, a value of *NO is used.
*NO  No IPDS pass-through is done.
*YES Specifies that IPDS pass-through is to be done if the spooled file is eligible for IPDS pass-through.

Note: Not all SCS or IPDS spooled files are eligible for IPDS pass-through. They may contain special functions that require transform to AFPDS for correct printing. Specifying IPDS pass-through on the printer file allows only those spooled files eligible for IPDS pass-through to bypass the extra transforms. Those spooled files not eligible for IPDS pass-through will still undergo the transforms to AFPDS and back to IPDS.

IPDS pass-through will not be valid for all PSF/400 supported printers. Any printer (or attachment) that does not support resident fonts can not support IPDS pass-through. This is because the resident font references in the data stream must be mapped to host fonts which are downloaded to the printer. All IBM IPDS printers, except for the following, can be supported with IPDS pass-through: 3820, 3825, 3827, 3828, 3829, 3831, 3835, 3900-001 and any printer attached to a system using the Distributed Print Function provided by either InfoPrint Manager or Print Services Facility for OS/2.

For V3R7, V4R1 and V4R2, IPDSPASTHR can be specified with the USRDFNDTA parameter in a printer file. You may continue using this support with existing printer files and PSF configuration objects by specifying IPDSPASTHR(*DEVD) in the printer file. If you specify a value of anything other than *DEVD for the IPDSPASTHR parameter, any IPDS pass-through value in the USRDFNDTA parameter is ignored.

User resource library list (USRRSCLIBL)

Specifies the list of user resource libraries to be used for searching for AFP resources for a spooled file. If the AFP resource is not found in the user resource libraries, then the library list specified in the DEVRSSCLIBL parameter of the PSF configuration object is searched. If no PSF configuration object is specified for the device, then libraries QFNTCPL, QFNT01-QFNT19, and QFNT61-69 are searched.

Single values
**DEVD**
The value specified for USRRSCLIBL in the PSF configuration object specified for a printer device description is used. If no PSF configuration object is specified for the device, a value of *JOBLIBL is used.

**NONE**
No user libraries are specified.

**JOBLIBL**
Specifies that the library list of the job that created the spool file is used in searching for AFP resources. This library list is saved with the spool file when it is created.

**CURLIB**
Specifies that the current library of the job that created the spool file is used for searching for AFP resources. If no library is specified as the current library for the job, then library QGPL is used.

Other values (up to 4 repetitions)

name Specify the name of a library that will be used to search for AFP resources. Up to four library names may be specified.

For V3R7, V4R1 and V4R2, USRRSCLIBL can be specified with the USRDFNDTA parameter in a printer file. PSF/400 uses that value if USRRSCLIBL(*PRTF) is specified in a PSF configuration object which is specified in the printer device description. You may continue using this support with existing printer files and PSF configuration objects by specifying USRRSCLIBL(*DEVD) in the printer file. If you specify a value of anything other than *DEVD for the USRRSCLIBL parameter, any user resource library value in the USRDFNDTA parameter is ignored.

Corner staple (CORNERSTPL)
Specifies the reference corner to be used for a corner staple. A staple is driven into the media at the reference corner. Refer to your printer’s documentation for information as to which reference corners are supported. Page rotation does not affect the placement of a corner staple.

**NONE**
A corner staple is not specified.

**DEVD**
The reference corner is the default reference corner used by the device.

**BOTRIGHT**
The reference corner is the bottom right corner of the media.

**TOPRIGHT**
The reference corner is the top right corner of the media.

**TOpleft**
The reference corner is the top left corner of the media.

**BOTLEFT**
The reference corner is the bottom left corner of the media.
Edge stitch (EDGESTITCH)

Specifies where one or more staples are driven into the media along the finishing operation axis. Refer to your printer’s documentation for information about which elements of this parameter are supported and which values for each element are supported. If specification of a value for an element is not supported by a printer, specify a value of *DEVD for that element. Page rotation does not affect the placement of an edge stitch.

Single values

*NONE
An edge stitch is not specified.

Element 1: Reference edge

Specifies the reference edge to be used for an edge stitch. An edge stitch is formed by having one or more staples driven into the media along the finishing operation axis.

*DEVD
The reference edge is the default reference edge used by the device.

*BOTTOM
The reference edge is the bottom edge of the media.

*RIGHT
The reference edge is the right edge of the media.

*TOP
The reference edge is the top edge of the media.

*LEFT
The reference edge is the left edge of the media.

Element 2: Reference edge offset

Specifies the offset of the edge stitch from the reference edge toward the center of the media.

*DEVD
The reference edge offset is the default reference edge offset used by the device.

0.0-57.79
Specify the offset of the edge stitch from the reference edge. If UOM(*CM) is specified, valid values range from 0 through 57.79, and if UOM(*INCH) is specified, valid values range from 0 through 22.75. This value is converted to millimeters for the printer. Fractional millimeters are not supported and are discarded when when conversion to millimeters is performed.

Element 3: Number of staples

Specifies the number of staples that are to be applied along the finishing operation axis.

*DEVD
The number of staples depends on the value of the Staple Offsets element of this parameter. If *DEVD is also specified or defaulted for the Staple Offsets element value, then the number of staples is the default number of staples used by the device. If one or more offsets are specified for Staple Offsets, the number of staples is the same as the number of staple offsets specified.

1-122
Specify the number of staples to be used for the edge stitch. If you specify the number of staples, then *DEVD must be specified for staple offsets. The device default for the spacing of each staple will be used.

Element 4: Staple offsets
Specifies the offset of the staples along the finishing operation axis. The offset is measured from the point where the finishing operation axis intersects either the bottom edge or the left edge of the media, toward the center of the media. Each consecutive value is used to position a single finishing operation centered on the specified point on the finishing operation axis.

Single values

*DEVD

The staple offsets are the default staple positions used by the device. If a value was specified for the Number of Staples element, the staple position of each staple will be calculated automatically by the printer.

Other values (up to 122 repetitions)

0.0-57.79

Specify the staple offset for each staple in the edge stitch. Up to 122 staple offsets may be specified. If one or more staple offsets values are specified, then *DEVD must be specified for the number of staples. If UOM(*CM) is specified, valid values range from 0 through 57.79, and if UOM(*INCH) is specified, valid values range from 0 through 22.75. This value is converted to millimeters for the printer. Fractional millimeters are not supported and are discarded when conversion to millimeters is performed.

Saddle stitch (SADLSTITCH)

Specifies where one or more staples are driven into the media along the finishing operation axis, which is positioned at the center of the media parallel to the reference edge. Refer to your printer's documentation for information about which elements of this parameter are supported and which values for each element are supported. If specification of a value for an element is not supported by a printer, specify a value of *DEVD for that element. Page rotation does not affect the placement of an edge stitch.

Single values

*NONE

A saddle stitch is not specified.

Element 1: Reference edge

Specifies the reference edge to be used for a saddle stitch. A saddle stitch is formed by having one or more staples driven into the media along the finishing operation axis, which is positioned at the center of the media parallel to the reference edge.

*DEVD

The reference edge is the default reference edge used by the device.

*TOP  The reference edge is the top edge of the media.

*LEFT  The reference edge is the left edge of the media.

Element 2: Number of staples

Specifies the number of staples that are to be applied along the finishing operation axis.

*DEVD

The number of staples depends on the value of the Staple Offsets element of this parameter. If *DEVD is also specified or defaulted for the Staple Offsets element value, then the number of staples is the default number of staples used by the device. If one or more offsets are specified for Staple Offsets, the number of staples is the same as the number of staple offsets specified.
1-122 Specify the number of staples to be used for the saddle stitch. If you specify the number of staples, then *DEVD must be specified for staple offsets. The device default for the spacing of each staple will be used.

Element 3: Staple offsets

Specifies the offset of the staples along the finishing operation axis. The offset is measured from the point where the finishing operation axis intersects either the bottom edge or the left edge of the media, toward the center of the media. Each consecutive value is used to position a single finishing operation centered on the specified point on the finishing operation axis.

Single values

*DEVD

The staple offsets are the default staple positions used by the device. If a value was specified for the Number of Staples element, the staple position of each staple will be calculated automatically by the printer.

Other values (up to 122 repetitions)

0.0-57.79

Specify the staple offset for each staple in the saddle stitch. Up to 122 staple offsets may be specified. If one or more staple offsets values are specified, then *DEVD must be specified for the number of staples. If UOM(*CM) is specified, valid values range from 0 through 57.79, and if UOM(*INCH) is specified, valid values range from 0 through 22.75. This value is converted to millimeters for the printer. Fractional millimeters are not supported and are discarded when conversion to millimeters is performed.

Font resolution for formatting (RNTRSL)

Specifies the resolution PSF/400 uses when printing to a multiple resolution printer configured to report multiple resolutions, but the spooled file does not specify the font metrics and resolution or the font is not available at the resolution that is contained in the spooled file.

For more information regarding the algorithm used for searching a library list for a font resource, see the Printer Device Programming manual section entitled User and Device Resource Library Lists in the chapter called Working With PSF configuration objects.

*DEVD

The value specified in the FNTRSL parameter of the PSF configuration object for the device is used. If no PSF configuration object is specified for the device, a value of *SEARCH is used.

*SEARCH

Specifies to search the library list for the first occurrence of a host font with a name match. The resolution of that font is used to print the spool file. Message PQT3546 is sent to specify the resolution of the font that was selected.

240 The font resolution is 240 pels per inch.

300 The font resolution is 300 pels per inch.

Defer write (DFRWRT)

Specifies whether output is held in the system buffer before being sent to the printer.
The system controls the amount of output that is held in the buffer before it is sent to the printer.

*YES  The system holds output in the buffer before sending it to the printer.

*NO   If *NO is specified for this parameter and *NO is specified for the Spool the data (SPOOL) parameter, output is not held in the buffer. Instead, output is sent immediately to the printer once the program has performed a write operation.

If *NO is specified for this parameter and *YES is specified for the SPOOL parameter, and if *IMMED is specified for the Spooled output schedule (SCHEDULE) parameter, output is held in the buffer until a page of output is available or until the system buffer is full.

If *IMMED is not specified for the SCHEDULE parameter, specifying *NO on this parameter has no effect.

Spool the data (SPOOL)

Specifies whether the output data for the printer device file is spooled. If *NO is specified, the other parameters on this command related to spooling are ignored.

*YES  The data is spooled.

*NO   The data is not spooled. It is sent to the printer device and printed as the output becomes available.

Spoool output queue (OUTQ)

Specifies, for spooled output files only, the name of the output queue for the printer file.

Single values

*JOB   The output queue specified in the job description is used.

*DEV   The output queue associated with the printer specified for the DEV parameter is used. The output queue has the same name as the printer.

Qualifier 1: Spooled output queue

name Specify the name of the output queue to which the output data is spooled.

Qualifier 2: Library

*LIBL  All libraries in the library list for the current thread are searched until the first match is found.

*CURLIB  The current job library list is used to locate the output queue. If no library is specified as the current library for the job, QGPL is used.

name Specify the library where the output queue is located.

Form type (FORMTYPE)

Specifies the type of form on which the output is printed. The identifiers used to indicate the type of forms are user-defined and can be a maximum of 10 characters in length.

*STD   The standard printer form for your computer system is used.
Specify the identifier of the form type used with this printer device file for printed output.

**Copies (COPIES)**

Specifies, for spooled output only, the number of copies of the output being printed.

- **1**: Only one copy of the output is printed.
- **1-255**: Specify the number of copies to print.

**Expiration date for file (EXPDATE)**

Specifies the expiration date for the spooled file. The spooled file will expire at 23:59:59, system local time on the date specified.

- ***NONE**: No expiration date is specified.
- ***DAYS**: The expiration date is to be calculated using the value specified for the **Days until file expires (DAYS)** parameter.

**date** Specify the date after which the spooled file will be eligible for removal from the system by the Delete Expired Spooled Files (DLTEXPSPLF) command. The date must be enclosed in apostrophes if date separator characters are used in the value.

**Days until file expires (DAYS)**

Specifies the number of days to keep the spooled file.

**Note**: A value must be specified for this parameter if the **Expiration date for file (EXPDATE)** parameter has a value of *DAYS. If the EXPDATE parameter has a value other than *DAYS, no value is allowed for this parameter.

- **1-366**: Specify an interval in days after which the spooled file will be eligible for removal from the system by the Delete Expired Spooled Files (DLTEXPSPLF) command. The actual expiration date applied to the spooled file is calculated by adding the number of days specified to the date the printer file is opened.

**Page range to print (PAGERANGE)**

Specifies, for spooled output files only, the starting and ending pages to print.

**Element 1: Starting page**

- **1**: Printing begins at page 1.
- ***ENDPAGE**: Use the end page value as the starting page.
integer
   Specify the starting page number.

Element 2: Ending page
*END  Printing continues until the end of the spooled file.
integer
   Specify the ending page number.

Max spooled output records (MAXRCDS)
Specifies, for spooled output only, the maximum number of records that can be in the spooled file for jobs using this printer file. If this maximum is reached, an inquiry message is sent to the program message queue.

100000  A maximum of 100000 records can be contained in the spooled output file for each job that uses this printer device file.

*NOMAX  There is no maximum on the number of records that can be in the spooled file.

1-999999  Specify the maximum number of records allowed.

File separators (FILESEP)
Specifies, for spooled output files only, the number of separator pages placed at the start of each printed file, including those between multiple copies of the same output.

0  No separator pages are used.
0-9  Specify the number of separator pages to be placed between printed files. If 0 is specified, no separator pages are printed for the file. In this case, the printed output for each file (or copy of a file) starts at the top of a new page.

Spooled output schedule (SCHEDULE)
Specifies, for spooled output files only, when the spooled output file is available to a writer.

*FILEEND  The spooled output file is available to the writer as soon as the file is closed.

*JOBEND  The spooled output file is available to the writer after the job is completed.

*IMMED  The spooled output file is made available to the writer as soon as the file is opened in the program.
**Hold spooled file (HOLD)**

Specifies, for spooled output only, whether the spooled file is held. The spooled file can be released by using the Release Spooled File (RLSSPLF) command.

- *NO* The spooled output file is not held by the output queue.
- *YES* The spooled output file is held until it is released by the Release Spooled File (RLSSPLF) command.

**Save spooled file (SAVE)**

Specifies, for spooled output files only, whether the spooled file is saved (left on the output queue) after the output has been produced.

- *NO* The spooled file data is not saved on the output queue.
- *YES* The spooled file data is saved on the output queue until the spooled file is deleted.

**Output priority (on OUTQ) (OUTPTY)**

Specifies the output priority for spooled output files that are produced by this job. The highest priority is 1 and the lowest priority is 9.

- *JOB* The output priority associated with the job that created the spooled file is used.
- 1-9 Specify a number ranging from 1 (high) through 9 (low).

**User data (USRDTA)**

Specifies, for spooled files only, some user-specified data that identifies the file.

- *SOURCE* If the spooled file was created by an application program, the name of the program is used. Otherwise, blanks are used.
  
  **character-value**
  
  Specify no more than 10 characters of text, enclosed in apostrophes.

**Spool file owner (SPLFOWN)**

Specifies, for spooled output only, who the owner of the spooled file is.

- *CURUSRPRF* The spooled file is owned by the current effective user of the current job or thread. See the Basic Printing information in the iSeries Information Center at http://www.ibm.com/eserver/iseries/infocenter for more detailed information on how the SPLFOWN parameter is affected when using any of the following APIs:
  - QWTSETP - Set Profile
- qsyssetuid() - Set User ID
- qsysseteuid() - Set Effective User ID
- qsysetreuid() - Set Real and Effective User ID

*JOB  The spooled file is owned by the original user profile of the job. If the job has switched to a new user profile, the original user profile is still the owner of the spooled file.

*CURGRPPRF  The spooled file is owned by the current effective group profile of the current job or thread. If there is no current effective group profile, ownership of the spooled file is determined in the same manner as *CURUSRPRF. See the Basic Printing information in the iSeries Information Center at http://www.ibm.com/eserver/iseries/infocenter for more detailed information on how the SPLFOWN parameter is affected when using any of the following APIs:
- QWTSETP - Set Profile
- qsyssetgid() - Set Group ID
- qsyssetegid() - Set Effective Group ID
- qsysetregid() - Set Real and Effective Group ID

*JOBGRPPRF  The spooled file is owned by the group profile of the original user profile of the job. If the job has switched to a new user profile, the group profile of the original user profile is still the owner of the spooled file. If no group profile exists, ownership of the spooled file is determined the same way as *JOB.

---

User Defined Option (USRDFNOPT)

Specifies, for spooled output only, one or more user-defined options to be used by user applications or user-specified programs that process spooled files. A maximum of four user-defined options can be specified.

Single values

*NONE  No user-defined options specified.

Other values (up to 4 repetitions)

character-value

Specify a user-defined option to be used by user applications or user-specified programs that process spooled files. All characters are acceptable.

---

User Defined Data (USRDFNDTA)

Specifies, for spooled output only, the user-defined data to be used by user applications or user-specified programs that process spooled files.

*NONE  No user-defined data specified.

character-value

Specify a user-defined data to be used by user applications or user-specified programs that process spooled files. All characters are acceptable.
User Defined Object (USRDFNOBJ)

Single values

*NONE  No user-defined object specified.

Element 1: Object

Qualifier 1: Object

name  Specify the user-defined object to be used by user applications or user-specified programs that process spooled files.

Qualifier 2: Library

*LIBL  All libraries in the library list for the current thread are searched until the first match is found.

*CURLIB  The current library for the thread is searched. If no library is specified as the current library for the thread, the QGPL library is searched.

name  Specify the name of the library to be searched.

Element 2: Object type

object-type  The user object type can be one of the following:

*DTAARA  Data Area

*DTAQ  Data Queue

*FILE  File

*PSFCFG  PSF Configuration Object

*USRIDX  User Index

*USRQ  User Queue

*USRSPC  User Space
DBCS character rotation (IGCCHRRTT)

Specifies whether the printer rotates double-byte character set (DBCS) characters 90 degrees counterclockwise when printing. The system prints rotated DBCS characters so that they appear in a vertical reading sequence. Alphanumeric characters are not rotated.

*NO  The system does not rotate DBCS characters when printing.
*YES  The system rotates DBCS characters 90 degrees counterclockwise when printing. The printer rotates each character individually.

DBCS characters per inch (IGCCPI)

Specifies the printer character density of double-byte character set (DBCS) characters, in characters per inch (CPI).

Note: This parameter does not specify the printer character density of alphanumeric characters. Alphanumeric characters are printed with the value specified for the CPI parameter.

*CPI   DBCS character density is based on the values specified for the Characters per inch (CPI) parameter. The system prints one double-byte character for every two alphanumeric characters.
   • For CPI(10), DBCS characters print at 5 characters per inch.
   • For CPI(12), DBCS characters print at 6 characters per inch.
   • For CPI(13.3), DBCS characters print at 6.7 characters per inch (same as IGCCPI(*CONDENSED)).
   • For CPI(15), DBCS characters print at 7.5 characters per inch.
   • For CPI(18), DBCS characters print at 9 characters per inch.
   • For CPI(20), DBCS characters print at 10 characters per inch.

5     DBCS character density is 5 characters per inch.
6     DBCS character density is 6 characters per inch.
10    DBCS character density is 10 characters per inch.

*CONDENSED
Condensed printing, in which the system prints 20 DBCS characters every 3 inches, is used. This value is valid for the 5553 or 5583 printers only.

DBCS SO/SI spacing (IGCSOSI)

Specifies how the system prints shift control characters.

*YES  The system prints shift control characters as blanks.
*NO   The system does not print shift control characters. These characters do not occupy a position on the printer output.
*RIGHT  The system prints two blanks when printing shift-in characters, but it does not print shift-out characters.
DBCS coded font (IGCCDEFNT)

Specifies the coded font that the system uses for double-byte character set (DBCS) printing.

Single values

*SYSVAL

The DBCS coded font specified in the system value QIGCCDEFNT is used.

Element 1: DBCS coded font

Qualifier 1: DBCS coded font

name Specify name of the DBCS coded font to use.

Qualifier 2: Library

*LIBL All libraries in the library list for the current thread are searched until the first match is found.

*CURLIB The current library for the job is used to locate the coded font name. If no library is specified as the current library for the job, the QGPL library is used.

name Specify the name of the library where the coded font name is located.

Element 2: Point size

*NONE

The point size is supplied by the system and is determined by the specified font character set.

0.1-999.9

Specify a point size.

Note: The point size parameter is only used when an outlined font is named, in other cases it is ignored.

Maximum file wait time (WAITFILE)

Specifies the number of seconds that the program waits for the file resources to be allocated when the file is opened, or the device or session resources to be allocated when an acquire operation is performed to the file. If the file resources cannot be allocated in the specified wait time, an error message is sent to the program.

*IMMED

The program does not wait. Immediate allocation of file resources is required.

*CLS The job default wait time is used as the wait time for the file resources to be allocated.

1-32767

Specify the number of seconds to wait for file resources to be allocated.
Share open data path (SHARE)

Specifies whether the open data path (ODP) is shared with other programs in the same routing step. When an ODP is shared, the programs accessing the file share facilities such as the file status and the buffer.

*NO  The ODP is not shared with other programs in the routing step. A new ODP for the file is created and used every time a program opens the file.

*YES The same ODP is shared with each program in the job that also specifies *YES when it opens the file.

Record format level check (LVLCHK)

Specifies whether the level identifiers of the record formats in the printer device file are checked when the file is opened by a program. If so, the record format identifiers in the program must match those in the device file. Because the same record format name can exist in more than one file, each record format is given an internal system identifier when it is created.

*YES  The level identifiers of the record formats are checked. If the level identifiers do not all match, an open error message is sent to the program requesting the open operation.

*NO  The level identifiers of the record formats are not checked.

Authority (AUT)

Specifies the authority you are giving to users who do not have specific authority for the object, who are not on an authorization list, and whose group profile or supplemental group profiles do not have specific authority for the object.

*LIBCRTAUT

The system determines the authority for the object by using the value specified for the Create authority (CRTAUT) parameter on the Create Library command (CRTLIB) for the library containing the object to be created. If the value specified for the CRTAUT parameter is changed, the new value will not affect any existing objects.

*CHANGE

The user can perform all operations on the object except those limited to the owner or controlled by object existence (*OBJEXIST) and object management (*OBJMGT) authorities. The user can change and perform basic functions on the object. *CHANGE authority provides object operational (*OBJOPR) authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove users.

*ALL  The user can perform all operations except those limited to the owner or controlled by authorization list management (*AUTLMGT) authority. The user can control the object’s existence, specify the security for the object, change the object, and perform basic functions on the object. The user also can change ownership of the object.

*USE  The user can perform basic operations on the object, such as running a program or reading a file. The user cannot change the object. Use (*USE) authority provides object operational (*OBJOPR), read (*READ), and execute (*EXECUTE) authorities.

*EXCLUDE

The user cannot access the object.
name Specify the name of an authorization list to be used for authority to the object. Users included in the authorization list are granted authority to the object as specified in the list. The authorization list must exist when the object is created.

Replace file (REPLACE)

Specifies whether an existing file, other than a save or database file, is replaced.

Note: The authority value for the file is determined by the user profile of the user. If an existing file is replaced, the authority value for the new file is copied from the replaced file.

*YES An existing file with the same name and library is replaced if the creation of the new printer device file is successful.

*NO The creation of a new printer device file is not allowed if there is an existing file with the same name and library.

Examples

Example 1: Creating a Printer File

CRTPRTF FILE(DSPHIST) SRCFILE(PRSNNL/JOBHIST) FILESEP(3)

This command creates a printer file named DSPHIST using the DDS source file named JOBHIST that is stored in the PRSNNL library. The defaults for the other parameters are assumed, except for FILESEP.

The printer uses standard forms that are 66 lines long and 132 print positions wide. An SCS data stream is used. It prints 6 lines per inch and overflows to a new page after line 60 is printed. The print image specified in the device description is used. Output is spooled to the output queue specified for the job and cannot be printed until the file is closed. The spooled file is not held or saved after printing. One copy of the output is printed, preceded by three separator pages, each containing the file name, the spooled number, and the job name and number. The print text specified in the current job is used.

Example 2: Creating a Printer File Containing DBCS Data

CRTPRTF FILE(IGCLIB/IGCPRT) IGCCA(*YES) FORMFEED(*AUTOCUT) IGCHRTT(*YES)

This command creates a printer file, IGCPRT (stored in library IGCLIB) that contains DBCS data. Cut sheets are automatically fed, and double-byte characters are rotated when printing.

Error messages

*ESCAPE Messages

CPF339F Expiration date must be today or a date in the future.

CPF7302 File &1 not created in library &2.
Create Proxy Command (CRTPRXCMD)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Create Proxy Command (CRTPRXCMD) command creates a new user-defined proxy command definition object. The proxy command definition object will have an object type of *CMD.

Once created, the proxy command can be changed by the Change Proxy Command (CHGPRXCMD) command. Other command-specific interfaces that reference a proxy command will actually operate on the command specified for the Target command (TGTCMD) parameter. The target command can be a regular command or another proxy command. Up to 5 proxy commands can be chained together. The last target command in the chain must be a regular command. Using the Change Command (CHGCMD) command or the Change Command Default (CHGCMD_DFT) command against the proxy command will change the target command. Prompting or running a proxy command will cause the target command to be prompted or run.

Other object operations not specific to commands will operate on the proxy command and will not affect the target command. For example, moving a proxy command to another library, saving a proxy command, or deleting a proxy command will only affect the proxy *CMD object; the target *CMD object is not affected.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMD</td>
<td>Command</td>
<td>Qualified object name</td>
<td>Required,</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Command</td>
<td>Name</td>
<td>Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>TGTCMD</td>
<td>Target command</td>
<td>Qualified object name</td>
<td>Required,</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Target command</td>
<td>Name</td>
<td>Positional 2</td>
</tr>
<tr>
<td>TEXT</td>
<td>Text ‘description’</td>
<td>Character value, *BLANK, *TGTCMD</td>
<td>Optional</td>
</tr>
<tr>
<td>REPLACE</td>
<td>Replace command</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
</tbody>
</table>

Command (CMD)

Specifies the proxy command to be created.

This is a required parameter.

Qualifier 1: Command

name Specify the name of the proxy command to be created.
Qualifier 2: Library

*CURLIB
The proxy command is created in the current library for the job. If no library is specified as the current library for the job, QGPL is used.

name Specify the name of the library where the proxy command is to be located.

Target command (TGTCMD)

Specifies the target command used to process the command. This command is not needed until the command is used in prompting, compiling, or running the proxy command. The parameters specified for the proxy command are passed to the target command for validation and interpretation. The target command can be a regular command or another proxy command. Up to 5 proxy commands can be chained together. The last target command in the chain must be a regular command.

This is a required parameter.

Qualifier 1: Target command

name Specify the name of the target command for this proxy command.

Qualifier 2: Library

*LIBL All libraries in the library list for the current thread are searched until the first match is found.

*CURLIB
The current library for the job is used to locate the command. If no library is specified as the current library for the job, QGPL is used.

*SYSTEM
Only the QSYS library is used to locate the command. If an exit program is registered for the QIBM_QCA_CHG_COMMAND exit point, the exit program will be allowed to change the command.

*NVLVLBL
Only the national language version (NLV) libraries in the library list and the QSYS library will be searched for the command. If an exit program is registered for the QIBM_QCA_CHG_COMMAND exit point, the exit program will be allowed to change the command.

name Specify the name of the library where the command is located.

Text ’description’ (TEXT)

Specifies the text that briefly describes the object.

*TGTCMD
The text description of the target command will be used as the text description of the proxy command. If the target command does not exist when this command is run, the text description of the proxy command will be the qualified name of the target command.

*BLANK
No text is specified.
'description'
Specify no more than 50 characters of text, enclosed in apostrophes.

**Authority (AUT)**

Specifies the authority you are giving to users who do not have specific authority for the object, who are not on an authorization list, and whose group profile or supplemental group profiles do not have specific authority for the object.

*USE The user can perform basic operations on the object, such as prompting or running the command. The user cannot change the object. Use (*USE) authority provides object operational (*OBJOPR), read (*READ), and execute (*EXECUTE) authorities.

*LIBCRTAUT
The system determines the authority for the object by using the value specified on the Create authority (CRTAUT) parameter on the Create Library command (CRTLIB) for the library containing the object to be created. If the value specified on the Create authority (CRTAUT) parameter is changed, the new value will not affect any existing objects.

*CHANGE
The user can perform all operations on the object except those limited to the owner or controlled by object existence (*OBJEXIST) and object management (*OBJMGT) authorities. The user can change and perform basic functions on the object. *CHANGE authority provides object operational (*OBJOPR) authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove users.

*ALL The user can perform all operations except those limited to the owner or controlled by authorization list management (*AUTLMGT) authority. The user can control the object’s existence, specify the security for the object, change the object, and perform basic functions on the object. The user also can change ownership of the object.

*EXCLUDE
The user cannot access the object.

name Specify the name of an authorization list. Users included on the authorization list are granted authority to the object as specified by the list. The authorization list must exist when the object is created.

**Replace command (REPLACE)**

Specifies whether or not an existing command object with the same name and library as the command being created should be replaced.

If you specify *YES and a command object already exists with the same name and in the same library, the existing command is renamed and moved to library QRPLOBJ, and will be deleted the next time an IPL of the operating system occurs.

*YES If the create operation is successful, the existing command is replaced by the new command.

*NO An existing command is not replaced, and the creation of a new command with the same name and library as an existing command is not allowed.
Examples

CRTPRXCMD CMD(QGPL/WJ) TGTMD(WRKJOB)
      TEXT('WRKJOB Shortcut')

The proxy command named WJ is created into library QGPL. When the WJ command is run or prompted, the target command WRKJOB in the library list will be used to process the WJ command parameters.

Error messages

*ESCAPE Messages

CPF0201
       Command &2 not created in library &3.
**Create PSF Configuration (CRTPSFCFG)**

**Where allowed to run:** All environments (*ALL)

**Threadsafe:** No

Use the Create PSF Configuration (CRTPSFCFG) command to create a Print Services Facility (PSF) configuration object from the information specified on this command.

A PSF configuration object allows you to specify additional parameters for an AFP printer that are not supported on the Create Device Description (Printer) CRTDEVPRT command, such as setting the device release timer. The object type for a PSF configuration object is *PSFCFG.

**Restrictions:**
- The PSF feature is required to use this command.
- You must have input/output system configuration (*IOSYSCFG) special authority to use this command.

### Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>PSFCFG</strong></td>
<td>PSF configuration</td>
<td>Qualified object name</td>
<td>Required,</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: PSF configuration</td>
<td>Name</td>
<td>Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *CURLIB</td>
<td></td>
</tr>
<tr>
<td><strong>USRRSCLIBL</strong></td>
<td>User resource library list</td>
<td>*JOBLIBL, *CURLIB, *NONE</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>DEVRSCLIBL</strong></td>
<td>Device resource library list</td>
<td>Single values: *DFT</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Other values (up to 30 repetitions): Name</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>IPDSPASTHR</strong></td>
<td>IPDS pass through</td>
<td>*NO, *YES</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>ACTRLSTMR</strong></td>
<td>Activate release timer</td>
<td>*NORDYF, *IMMED, *PRTNORDYF, *PRTIMMED</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>RESTRTRMR</strong></td>
<td>Restart timer</td>
<td>1-1440, *IMMED</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>RETRY</strong></td>
<td>APPC and TCP/IP retry count</td>
<td>1-99, 15, *NOMAX</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>RETRYDLY</strong></td>
<td>Delay between APPC retries</td>
<td>0-999, 90</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>ACKFRQ</strong></td>
<td>Acknowledgment frequency</td>
<td>1-32767, 100</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>PRTRSPTRMR</strong></td>
<td>Printer response timer</td>
<td>5-3600, *NOMAX</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>PDFGEN</strong></td>
<td>Generate PDF output</td>
<td>Single values: *NONE</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Other values (up to 3 repetitions): *SPLF, *STMF, *MAIL</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>PDFDEVTYPE</strong></td>
<td>PDF device emulation type</td>
<td>*IPA40240, *IPA40300, *IPA4028, *P3812</td>
<td>Optional</td>
</tr>
</tbody>
</table>

© Copyright IBM Corp. 1998, 2006
<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>PDFMULT</td>
<td>Multiple PDF files</td>
<td>Single values: *NO Other values: Element list</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Element 1: Acknowledge</td>
<td>*YES</td>
<td></td>
</tr>
<tr>
<td></td>
<td>multiple groups</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Process option</td>
<td>*SPLIT, *INDEX</td>
<td></td>
</tr>
<tr>
<td>PDFINCFNT</td>
<td>PDF fonts inline</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
<tr>
<td>PDFDTAQ</td>
<td>PDF data queue</td>
<td>Single values: *NONE Other values: Qualified object name</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: PDF data queue</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td>PDFMAILSVR</td>
<td>PDF mail server name</td>
<td>Single values: *SNDDST Other values (up to 4 repetitions): Character value, *LOCAL</td>
<td>Optional</td>
</tr>
<tr>
<td>PDESENDER</td>
<td>Sender of electronic mail</td>
<td>Name, *SPLFOWN, QSPLJOB</td>
<td>Optional</td>
</tr>
<tr>
<td>PDFADMIN</td>
<td>PDF administrator</td>
<td>Character value, *NONE</td>
<td>Optional</td>
</tr>
<tr>
<td>PDFMAPPGM</td>
<td>PDF user program</td>
<td>Single values: *NONE, *IBMPGM Other values: Qualified object name</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: PDF user</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>program</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td>PDFMAP</td>
<td>PDF mapping object</td>
<td>Single values: *NONE Other values: Qualified object name</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: PDF mapping</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>object</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td>PDFOUTQ</td>
<td>PDF output queue</td>
<td>Qualified object name</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: PDF output</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>queue</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td>PDFDIR</td>
<td>PDF directory</td>
<td>Character value</td>
<td>Optional</td>
</tr>
<tr>
<td>AFPSAVE</td>
<td>Save AFP data</td>
<td>*NO, *YES</td>
<td>Optional</td>
</tr>
<tr>
<td>AFPDATABASE</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>AFPOUTQ</td>
<td>AFP output queue</td>
<td>Qualified object name</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: AFP output</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>queue</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td>TEXT</td>
<td>Text ‘description’</td>
<td>Character value, *BLANK</td>
<td>Optional</td>
</tr>
<tr>
<td>AUTOSSNRCY</td>
<td>Automatic session recovery</td>
<td>Single values: *NO Other values: Element list</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Element 1: Enabled</td>
<td>*YES</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Message option</td>
<td>*INFO, *INQ</td>
<td></td>
</tr>
<tr>
<td>BLANKPAGE</td>
<td>Blank page</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
<tr>
<td>PAGSIZCTRL</td>
<td>Page size control</td>
<td>*NO, *YES</td>
<td>Optional</td>
</tr>
<tr>
<td>RESFONT</td>
<td>Resident fonts</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
<tr>
<td>RSCRET</td>
<td>Resource retention</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
<tr>
<td>EDGEORIENT</td>
<td>Edge orient</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
<tr>
<td>USEOUTLNT</td>
<td>Use outline fonts</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
<tr>
<td>PSDFNOPT</td>
<td>PSF defined option</td>
<td>Single values: *NONE Other values (up to 6 repetitions): Character value, *NONE</td>
<td>Optional</td>
</tr>
<tr>
<td>FNTSUBMSG</td>
<td>Font substitution messages</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
<tr>
<td>FNTCAPTURE</td>
<td>Capture host fonts at printer</td>
<td>*NO, *YES</td>
<td>Optional</td>
</tr>
<tr>
<td>Keyword</td>
<td>Description</td>
<td>Choices</td>
<td>Notes</td>
</tr>
<tr>
<td>-----------</td>
<td>------------------------------------------</td>
<td>------------------------------</td>
<td>----------</td>
</tr>
<tr>
<td>FNTRSL</td>
<td>Font resolution for formatting</td>
<td>*SEARCH, 240, 300</td>
<td>Optional</td>
</tr>
<tr>
<td>FNTTBL</td>
<td>Font mapping table</td>
<td>Single values: *NONE</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Other values: Qualiﬁed object name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualiﬁer 1: Font mapping table</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualiﬁer 2: Library</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td>CSEMODE</td>
<td>Cut sheet emulation mode</td>
<td>*NONE, *CHKFIRST, *CHKALL</td>
<td>Optional</td>
</tr>
<tr>
<td>MAPGCGFNT</td>
<td>Use DBCS simulation fonts</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
<tr>
<td>REPLACE</td>
<td>Replace</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
</tbody>
</table>

**PSF configuration (PSFCFG)**

Specifies the Print Services Facility (PSF) configuration object to be created.

This is a required parameter.

Qualifier 1: PSF configuration

*name* Specify the name of the PSF configuration object to be created.

Qualifier 2: Library

* CURLIB  
  Store the PSF configuration object in the current library. If no library is specified as the current library for the job, the QGPL library is used.

  *name* Specify the name of the library where you want to store the PSF configuration object.

**User resource library list (USRRSCLIBL)**

Specifies the user resource library list to use when searching for AFP resources. When searching for an AFP resource specified with a spooled file, Print Services Facility (PSF) first searches the libraries in the user resource library list and then those in the device resource library list.

*PRTF* has been removed as a valid value for the USRRSCLIBL parameter. PSF configuration objects migrated from other releases that were created with USRRSCLIBL(*PRTF*) will be supported exactly as in prior releases as long as the new USRRSCLIBL parameter on the printer file has the a value of *DEVD*. When a CHGPSFCFG command is run in this environment, a value of *SAME* will be displayed where a value of *PRTF* would have been displayed on a prior release.

* JOBLIBL  
  Use the library list for the job that created the spooled file when searching for AFP resources.  
  Each time the user creates a new spooled file, the job library list at that point in time is saved.

* CURLIB  
  Use the current library for the job that created the spooled file when searching for AFP resources.  
  If no library is specified as the current library for the job, then the QGPL library is used.
*NONE

No user resource library list is used for searching for AFP resources. Only the device resource library list is used.

---

**Device resource library list (DEVRSCLIBL)**

Specifies the device resource library list to use when searching for AFP resources. When searching for an AFP resource specified with a spooled file, Print Services Facility (PSF) first searches the libraries in the user resource library list and then the libraries in the device resource library list.

**Single value**

*DFT*  
PSF searches these libraries, if they exist, when searching for AFP resources:

- QFNTCPL
- QFNT01 - QFNT19
- QFNT61 - QFNT69

**Note:** If not all the system libraries in the above list have been created, a user can create libraries using the names of the missing system libraries. If this occurs and you specify *DFT on the DEVRSCLIBL parameter, the resources in those user-created libraries could be mistakenly found by other users. To prevent this, the system administrator should create all of the missing system libraries with PUBLIC *USE authority.

**Other values**

*name*  
Specify up to 30 names of libraries PSF will use to search for AFP resources.

---

**IPDS pass through (IPDSPASTHR)**

Specifies whether IPDS pass-through is done for the device. IPDS pass-through is a mechanism by which unnecessary datastream conversions can be eliminated, thus improving throughput and decreasing CPU utilization. Full page-level error recovery is supported.

IPDS pass-through can be used for SCS and IPDS files which do not specify any AFP processing features, such as a front or back overlay on the printer file. SCS data is transformed to a generic IPDS. Specifying IPDS pass-through on the device configuration or printer file allows only those spooled files eligible for IPDS pass-through to bypass the extra transforms. Those spooled files not eligible for IPDS pass-through will still undergo the transforms to AFPDS and back to IPDS.

*NO*  
No IPDS pass-through is done.

*YES*  
IPDS pass-through is performed for the device for all spooled files that are eligible for IPDS pass-through.

IPDS pass-through is not valid for all Print Services Facility (PSF) supported printers. Only printers that support resident fonts can be used with IPDS pass-through. If a printer does not support resident fonts, font references in the data stream must be mapped to host fonts, which are then downloaded to the printer. This requires the transform to AFPDS and back to IPDS. The following IPDS printers cannot support IPDS pass-through:

- 3820, 3825, 3827, 3828, 3829, 3831, 3835, 3900-001
• Any Distributed Print Function (DPF)-attached printer. DPF is a function supported by Infoprint Manager for Windows NT and Windows 2000, which blocks the use of printer-resident fonts.

Activate release timer (ACTRLSTMR)

Specifies the point at which the release timer is activated. The value specified for Release timer (RLSTMR) determines the length of time the writer will "keep" the printer before releasing the session.

*NORDYF

The release timer is activated when there are no ready (RDY) spooled files in the printer’s output queue and the last page of the last spooled file processed has printed. If the release timer expires, the session to the printer is released but the writer does not end. When the session is released, another Print Services Facility (PSF) can start a session to the printer.

Use this value when you want the writer to print all ready spooled files before releasing the session.

*NORDYF is supported only for printers and devices attached to the system using APPC or TCP/IP. For an APPC connection, use this value only with the PSF Direct support provided by Infoprint Manager for AIX or Infoprint Manager for Windows NT and Windows 2000. For a TCP/IP connection, this value can be used for any printer. This value is not supported for twinaxial-attached printers.

*IMMED

The release timer is activated immediately after PSF has successfully linked to the printer. If the release timer expires, the session to the printer is released but the writer does not end. If a file is being printed when the release timer expires, the writer releases the session after all pages of the spooled file have printed. When the session is released, another PSF can start a session to the printer.

Use this value when you want the writer to share the printer with another print writer.

*IMMED is supported only for printers and devices attached to the system using APPC or TCP/IP. For an APPC connection, use this value only with the PSF Direct support provided by Infoprint Manager for AIX or Infoprint Manager for Windows NT and Windows 2000. For a TCP/IP connection, this value can be used for any printer. This value is not supported for twinaxial-attached printers.

*PRTNORDYF

This value can be specified if you are using a printer that allows control over the exchange of IPDS data (the IPDS dialog). Refer to Printer Information, 5544-5750 to determine if your printer supports this feature.

This value specifies that the release timer is to be activated after all of these conditions are met:

• The writer receives an indication from the printer to release the IPDS dialog.
• There are no ready spooled files in the printer’s output queue.
• The last page of the last spooled file processed has printed.

If the release timer expires, the writer releases the IPDS dialog with the printer. The session is not released and the port in use by the writer is not available to another PSF. Another printer driver can start a dialog with the printer on a different printer port.

If the writer detects that the printer is not capable of controlling the IPDS dialog, then the value is ignored and PSF behaves as if RLSTMR(*NOMAX) was specified.

Use this value when you want the writer to print all ready spooled files with before releasing the IPDS dialog.
*PRTNORDYF is supported on a twinaxial, TCP/IP or APPC connection.

*PRTIMMED
This value can be specified if you are using a printer that allows control over the exchange of IPDS data (the IPDS dialog). Refer to Printer Information, S544-5750 to determine if your printer supports this feature.

This value specifies that the release timer is to be activated immediately after the writer receives an indication from the printer to release the IPDS dialog. If the release timer expires, the writer releases the IPDS dialog with the printer, but the session is not released. The port in use by the writer is not available for use by another PSF, however another printer driver can start a dialog with the printer on a different printer port. If a file is being printed when the release timer expires, the writer releases the dialog after all pages of the spooled file have printed.

Use this value when you want to specify the length of time the writer controls the printer after the printer has indicated that it is needed by a printer driver at another printer port.

If the writer detects that the printer is not capable of telling the writer to stop the flow of data, then this value is ignored, and PSF behaves as if RLSTMR(*NOMAX) was specified.

*PRTIMMED is supported on a twinaxial, TCP/IP or APPC connection.

---

**Release timer (RLSTMR)**

Specifies the amount of time to wait after the release timer has been activated and the last page of the last ready spooled file has printed before releasing the printer. Print Services Facility (PSF) does not end, but releases the connection or IPDS dialog with the printer. See the Activate release timer (ACTRLSTMR) parameter description for additional information on the release timer.

When a spooled file becomes ready, PSF attempts to establish a session with the printer. See the Restart timer (RESTRTMR) parameter description for additional information.

*NOMAX
The printer is not released unless the End Writer (ENDWTR) command is run.

*SEC15
PSF waits 15 seconds before releasing the printer.

*SEC30
PSF waits 30 seconds before releasing the printer.

1-1440 Specify the number of minutes the printer writer waits before releasing the printer or IPDS dialog.

---

**Restart timer (RESTRTMR)**

Specifies the amount of time to wait before the printer writer attempts to re-establish either a session or dialog. To determine whether a session or dialog is to be re-established, the printer writer considers the following:

- The value specified for ACTRLSTMR.
- Whether the printer supports IPDS dialog management.
- The type of link: twinaxial, APPC or TCP/IP.

If ACTRLSTMR(*NORDYF) or ACTRLSTMR(*IMMED) are specified, the session is restarted if the printer is attached using APPC or TCP/IP.
If ACTRLSTMR(*PRTNORDYF) or ACTRLSTMR(*PRTIMMED) are specified, the dialog is restarted if the printer supports dialog management.

See the **Activate release timer** (ACTRLSTMR) parameter description for additional information about session and dialog management control.

**IMMED**

The printer writer attempts to re-establish the session or dialog as soon as a spooled file has a status of RDY.

**1-1440** Specify the number of minutes the printer writer waits, after a session or dialog have been released and a spooled file has a status of RDY, before attempting to connect.

---

### APPC and TCP/IP retry count (RETRY)

Specifies the number of times to retry a session start request when attempting to establish a session with a printer. This parameter applies to printers and devices configured for either TCP/IP or APPC.

**15** Fifteen retry attempts are made to establish a session. If after fifteen retries Print Services Facility (PSF) still cannot establish a session, the printer writer ends.

**NOMAX**

No limit is put on the number of retries. PSF continues issuing session start requests until the session is established or the printer writer is ended using ENDWTR OPTION(*IMMED).

**1-99** Specify the number of retry attempts to establish a session.

---

### Delay between APPC retries (RETRYDLY)

Specifies the number of seconds Print Services Facility (PSF) pauses after it receives notification that a session start request has failed. After the specified time has elapsed, another session start request is issued. The number of retries performed by PSF is controlled by parameter RETRY. This parameter applies to printers and devices configured for APPC.

**90** A 90-second delay will be used between retry attempts.

**0-999** Specify the number of seconds to pause between retry attempts to establish a session.

---

### Acknowledgment frequency (ACKFRQ)

Specifies the frequency, in pages, with which Print Services Facility (PSF) sends IPDS acknowledgment requests to a printer. The acknowledgment request responses from the printer contain information about the status of pages sent to the printer.

If a spooled file contains fewer pages than specified for ACKFRQ, an acknowledgment is requested after the last page of the spooled file is sent.

Consider adjusting this value when specifying AUTOSSNRCY(*YES). When a connection with a printer is abnormally ended, PSF may reprint pages because the printer was unable to return the status of pages printed. By increasing the frequency with which acknowledgments are sent, the number of pages which might be reprinted is decreased when a severed connection is restored. However, if acknowledgments are requested with great frequency, such as once per page, you may notice a performance degradation.
Acknowledgment frequency is supported on all attachments: twinaxial, APPC and TCP/IP. Note that AUTOSSNRCY is supported on APPC and TCP/IP attachments only.

| 100  | Specifies that an acknowledgment request is sent to the printer after every 100 pages. |
| 1-32767 | Specifies the number of pages after which PSF sends an acknowledgment request to the printer. |

**Printer response timer (PRTRSPTMR)**

Specifies the time, in seconds, to wait for a response from a TCP/IP attached printer.

- *NOMAX
  - The printer writer will wait for a response from the printer until one is received. If the writer does not receive a message, it is never ended.
- 5-3600
  - Specifies the time, in seconds, the printer writer should wait for a response from the printer. The writer is ended if the printer does not respond within the specified amount of time. If this happens, the writer ends and a message is sent to the message queue.

**Generate PDF output (PDFGEN)**

Specifies whether to generate a PDF output file through an IPDS to PDF transform when processing the spooled file. You can spool the generated PDF file, store it as a stream file, send it as electronic mail, or any combination of those. You must have Infoprint Server installed to support this feature.

In order to generate PDF, the remote location name for the printer device description must either be a valid loopback address or a name associated with a valid loopback address. An Internet address representing a valid loopback address must have 127 as the first octet of the Internet address.

**Single value**

- *NONE
  - Do not generate a PDF output file.

**Other values (up to 3 repetitions)**

- *SPLF
  - Place the generated PDF output file in a spooled file. The value specified for the PDF output queue (PDFOUTQ) parameter identifies the output queue to be used.
- *STMF
  - Place the generated PDF output file in a stream file. The value specified for the PDF directory (PDFDIR) parameter identifies the integrated file system (IFS) directory to be used.
- *MAIL
  - Electronically mail the PDF output file.

**PDF device emulation type (PDFDEVTYPE)**

Specifies the type of device that the IPDS to PDF transform’s virtual printer should emulate.

- *IP40240
  - Emulate an IP40 printer device configured at 240 pel resolution.
*IP40300
   Emulate an IP40 printer device configured at 300 pel resolution.
*4028  Emulate a 4028 printer device.
*3812  Emulate a 3812 printer device.

PDF paper size drawer 1 (PDFPPRDWR1)
Specifies the size of paper in drawer one of the device associated with the IPDS to PDF transform. This information is used to determine the generated PDF page size.

*LETTER
   North American letter size media (8.5 x 11 inches).
*LEGAL
   North American legal size media, (8.5 x 14 inches).
*STATEMENT
   North American statement size media (5.5 x 8.5 inches).
*EXECUTIVE
   North American executive size media (7.5 x 10.5 inches).
*LEDGER
   North American ledger size media (11 x 17 inches).
*A5    ISO A5 size media (148.5 x 210 mm).
*A4    ISO A4 size media (210 x 297 mm).
*A3    ISO A3 size media (297 x 420 mm).
*B5    ISO B5 size media (176 x 250 mm).
*B4    ISO B4 size media (257 x 364 mm).

PDF paper size drawer 2 (PDFPPRDWR2)
Specifies the size of paper in drawer two of the device associated with the IPDS to PDF transform. This information is used to determine the generated PDF page size.

*LETTER
   North American letter size media (8.5 x 11 inches).
*LEGAL
   North American legal size media, (8.5 x 14 inches).
*STATEMENT
   North American statement size media (5.5 x 8.5 inches).
*EXECUTIVE
   North American executive size media (7.5 x 10.5 inches).
*LEDGER
   North American ledger size media (11 x 17 inches).
*A5    ISO A5 size media (148.5 x 210 mm).
*A4    ISO A4 size media (210 x 297 mm).
Multiple PDF files (PDFMULT)

Specifies the action the IPDS to PDF transform should take when encountering multiple groups within the input data.

Single value
*NO   Ignore group boundaries and create a single output file.

Element 1: Acknowledge multiple groups
*YES   Process multiple groups with the IPDS to PDF transform based on the value specified for element 2 of this parameter.

Element 2: Process option
*SPLIT Multiple PDF output files will be generated. The file will be split at group boundaries.

*INDEX An index tag or bookmark will be placed at the group boundaries in a single output file.
   If *INDEX is requested, bookmarks will be labeled according to:
   • the group name on the DDS STRPAGGRP keyword or
   • index entries generated by CRTAFPDTA or
   • BNG tags inserted by Toolbox
   If characters in the group name are not available in the standard PDF encoding they will be presented as a space.

PDF fonts inline (PDFINCFNT)

Specifies whether the PDF output generated by the IPDS to PDF transform carries the necessary fonts inline. Including the fonts inline guarantees font fidelity but increases the PDF file size.

If the user chooses not to have the fonts embedded, the IPDS Type 1 font name character string is moved to the PDF font controls. When the document is viewed the Acrobat Reader will map IBM’s core font names to the equivalent Adobe or client environment set of core fonts. For any font name character strings that Adobe Acrobat does not have an equivalent for, Adobe Acrobat will use the Adobe multi-master font substitution program to select the available font that will constitute the "best fit".

*YES   The fonts should be carried inline with the PDF output.
*NO    The fonts should not be carried inline with the PDF output.
PDF data queue (PDFDTAQ)

Specifies the name of the data queue where Print Services Facility (PSF) will log the IPDS to PDF transformation completion notifications.

This parameter is optional, but if a data queue is specified, the data queue must exist when this command is run.

Single value

*NONE
IPDS to PDF transformation completion notifications will not be logged to any data queue.

Qualifier 1: PDF data queue

ame 
Specify the name of the data queue to be used.

Qualifier 2: Library

name
Specify the name of the library where the data queue is located.

PDF mail server name (PDFMAILSVR)

Specifies which mail server to use for electronically mailing the resulting PDF file from the IPDS to PDF transform. This parameter is only valid if PDFGEN(*MAIL) is specified.

Single value

*SNDDST
Use the Send Distribution (SNDDST) command to e-mail the PDF output.

Other values (up to 4 repetitions)

*LOCAL
Use the local machine as the mail server. The SMTP protocol is used for sending the e-mail. You can specify *LOCAL in any position in the list of mail servers.

character-value
Specify the domain name or Internet address of the mail server to use to electronically mail the PDF output.

You can specify up to 4 mail servers. The writer uses the mail servers in the order in which they are listed. If the writer detects that the first mail server cannot be used, the writer will attempt to use the additional servers that have been specified in the list. Print Services Facility (PSF) internally reorganizes the list of servers, making certain to always first attempt to use the last server that was working.

If no usable server can be found in the list, the action taken is determined by the value specified for the PRTRRMSG parameter of the printer device description. If PRTRRMSG(*INFO) is specified, the writer is ended. If PRTRRMSG(*INQ) is specified, then an inquiry message is issued.

Sender of electronic mail (PDFSENDER)

Specifies the name to use as the sender for a PDF file sent by electronic mail.
*SPLFOWN

Print Services Facility (PSF) uses the user profile for the spooled file’s owner to obtain the sender of the electronic mail.

QSPLJOB

The electronic mail is being sent from PSF.

name Specify a valid user profile. PSF uses this user profile to obtain the sender of the electronic mail.

When PSF uses a user profile to determine the electronic mail sender, the user profile must exist on the system and must have an entry in the System Distribution Directory with a user ID specified. If you are using an SMTP mail server to send the mail, the directory entry must also have an SMTP user ID. If the profile has an SMTP User ID, that User ID is used as the sender, even if you use SNDDST to send the mail. Otherwise, if you use SNDDST to send the mail and there is no SMTP User ID, the profile’s User ID is used.

For example, if user profile MY_PROFILE has a User ID of JIM and an SMTP User ID of JIMJ, the electronic mail sender is JIMJ, regardless of the mail server used. If the user profile had no SMTP User ID and you use SNDDST to send the mail, the electronic mail sender is JIM.

PDF administrator (PDFADMIN)

Specifies the e-mail address for the designated PDF administrator. The administrator will be notified when files cannot be delivered to the designated destination. Not all failures will be recoverable, as some errors occur after control of the delivery has passed to other components of the system. For example, the PDF administrator is not notified of undeliverable e-mails.

*NONE

No PDF administrator is specified. If e-mail notification was to be sent to a PDF administrator, the notification will not be sent.

'character-value'

Specify no more than 80 characters of text that constitutes a valid e-mail address, enclosed in apostrophes.

PDF user program (PDFMAPPPGM)

Specifies the name of a mapping program that Print Services Facility (PSF) will call to customize the PDF transform, such as specifying encryption or, when the PDF is being sent as e-mail, resolving one or more mail tags in the spooled file. If a mapping program is not specified, PSF assumes that the mail tag is a valid electronic mail address and will attempt to send the file using the information in the mail tag.

The PSF configuration object will not be created if the mapping program specified does not exist. If the mapping program is deleted before the spooled file is processed, the PDF output file will be deleted, an error message will be issued to the message queue associated with the printer writer, and the original spooled file will be held.

Single values

*NONE

No mapping program will be used to customize the PDF transform.
*IBMPGM

The default IBM-supplied mapping program will be used to customize the PDF transform. If the value *IBMPGM is specified, you must also specify a value for the PDFMAP parameter.

Qualifier 1: PDF user program

ame Specify the name of the user mapping program to be used.

Qualifier 2: Library

ame Specify the name of the library where the user mapping program is located.

PDF mapping object (PDFMAP)

Specifies the name of a mapping object that Print Services Facility (PSF) will pass to the PDF mapping program. If no mapping object is specified, PSF assumes that the mapping program does not require one. A mapping object is required if *IBMPGM is specified as the mapping program on the PDFMAPPGM parameter. Also, if you specify a mapping object, you must specify *IBMPGM on the PDFMAPPGM parameter.

The PSF configuration object will not be created if the mapping object specified does not exist. If the mapping object is deleted before the spooled file is processed, the PDF output file will be deleted, an error message will be issued to the message queue associated with the printer writer, and the original spooled file will be held.

Single value

*NONE

No mapping object will be passed to the specified mapping program to resolve file destination.

Qualifier 1: PDF mapping object

name Specify the name of the user mapping object.

Qualifier 2: Library

name Specify the name of the library where the user mapping object is located.

PDF output queue (PDFOUTQ)

Specifies the output queue to be used when *SPLF is specified for the Generate PDF output (PDFGEN) parameter. A value is required for this parameter when *SPLF is specified for the PDFGEN parameter.

Qualifier 1: PDF output queue

name Specify the name of the output queue to be used.

Qualifier 2: Library

name Specify the name of the library where the output queue is located.
PDF directory (PDFDIR)

Specifies the path where the PDF file should be stored. An integrated file system directory must be specified when *STMF is specified for the PDFGEN parameter. The name must begin with a / . The directory name in the path cannot contain any of the following characters: \ < > " ? : * |

The subdirectories and files created by the transform will be owned by the original spooled file owner and will have public authority of *EXCLUDE.

If you want to e-mail this file later, specify QDLS/directory-name for the PDF directory. This will cause your file to be stored in the following location:

/QDLS/directory-name/job-name/job-number/job-user-name/file-number/date/sequence-number/

For storage in the root file system, simply specify the subdirectory where the file should be stored (directory-name). This will cause your file to be stored in the following location:

/directory-name/job-name/job-number/job-user-name/job-number_file-number_date_sequence-number/

Note: The last subdirectory is a concatenation of a number of values to guarantee a unique file name.

The file name components are explained below:

- **job-name** The ten character job name of the original spooled file
- **job-number** The job number assigned to the original spooled file, prefixed by the last two characters of the job name
- **job-user-name** The ten character name of the original spooled file owner
- **file-number** The file number of the original spooled file, prefixed by the last two characters of the job-user-name
- **date** The two digit month appended to the two digit day appended to the four digit year when the PDF transform completed
- **sequence-number** A six character sequence number. It will be set to 000001 if PDFMULT is *NO. If PDFMULT is *YES, the sequence number is incremented to uniquely identify each PDF file generated for the job.

'character-value'

Specify the name of the integrated file system (IFS) directory to be used.

Save AFP data (AFPSAVE)

Specifies whether the Print Services Facility (PSF) product should activate the capability to retain the generated AFPDS file on an output queue upon completion of processing. This parameter specifies that PSF should generate an AFPDS file from an SCS, AFPDS, IPDS, PostScript, PCL, or PDF input data stream and place the AFPDS on an output queue. The output queue will be determined by a user exit program in the case of segmented print requests, or by the value provided by the AFPOUTQ parameter in the PSF configuration object. The AFPOUTQ parameter is required if AFPSAVE is set to *YES.

This setting will be ignored for input data streams of line and mixed mode data and when IPDS passthrough is active. You must have Infoprint Server installed to support PostScript, PCL, and PDF input data streams.
Note: If you are not processing segmented print requests using a PDF mapping program, you must also specify a value for the AFPRESPOOL argument to the USRDFNDTA parameter when submitting your print request for this function to be performed. Refer to Printer Device Programming, SC41-5713 for more information on using the USRDFNDTA parameter.

*NO  The file should not be saved after processing has been completed.

*YES  The generated AFPDS file may be saved after processing has been completed. For the AFPDS file to be saved, the input data stream must support the saving of AFPDS data and must meet one of these conditions:
  • The print request is segmented and the PDF mapping program requests that the segment be respooled.
  • The print request is not segmented and the user has specified the AFPRESPOOL argument in the USRDFNDTA parameter on the print request.

AFP output queue (AFPOUTQ)

Specifies the output queue to use when *YES is specified for the Save AFP data (AFPSAVE) parameter. A value is required for this parameter when *YES is specified for the AFPSAVE parameter. Segmented print requests using a PDF mapping program can override this value.

Qualifier 1: PDF output queue

name  Specify the name of the output queue to use.

Qualifier 2: Library

name  Specify the name of the library where the output queue is located.

Text 'description' (TEXT)

Specifies the text that briefly describes the Print Services Facility (PSF) configuration object.

*BLANK  No text is specified.

'text description'
  Specify no more than 50 characters of text, enclosed in apostrophes.

Automatic session recovery (AUTOSSNRCY)

Specifies whether Print Services Facility (PSF) will automatically attempt to resume printing when a session has been unexpectedly ended by a device. This parameter applies to devices configured in a printer device description for an APPC or TCP/IP attachment.

Single value

*NO  Specifies that PSF ends when a session has been unexpectedly ended by a device.

Element 1: Enabled
*YES Specifies that PSF attempts to re-establish a session which has been unexpectedly ended by a device.

If you are using a printer device description which specifies an APPC attachment, note the following:
- PSF configuration object parameters RETRY and RETRYDLY are used when PSF is attempting to re-establish a session.
- You must use an APPC controller description and an APPC device description that specifies APPN(*YES). Additionally, the APPC controller description must have MINSWTSTS(*VRYONPND) specified.

If you are using a printer device description which specifies a TCP/IP connection, then PSF configuration object parameter RETRY is used when PSF is attempting to re-establish a session.

To avoid reprinting pages, you may want to specify *INQ for this parameter’s second element or decrease the value specified for the Acknowledgment frequency (ACKFRQ) parameter.

If you decrease the Acknowledgment frequency, PSF will be able to track printed pages more closely. However, there could be some degradation in performance. This depends on your network and your perception of the performance.

If you select inquiry message notification (*INQ), then you can be very specific about the page at which printing should resume.

Element 2: Message option

*INFO An informational message is sent to the message queue associated with the writer when PSF is performing automatic session recovery.

*INQ An inquiry message is sent to the message queue associated with the writer when PSF is performing automatic session recovery. This message lets you specify the page number from which a writer should begin printing the last spooled file being processed.

Blank page (BLANKPAGE)

Specifies whether Print Services Facility (PSF) issues a blank page after every separator page and spooled file copy that contains an odd number of pages. The blank pages assure that the printer output is placed into the output stacker in a manner suitable for bursting. This parameter only applies to the following continuous forms printers:
- 3831
- 3835
- 3900-001
- All AFCCU continuous forms printers.

*YES PSF issues a blank page after every separator page and spooled file copy that contains an odd number of pages.

*NO PSF does not issue a blank page after every separator page and spooled file copy that contains an odd number of pages.
Page size control (PAGSIZCTL)

Specifies whether the page size (forms) in the printer is set by Print Services Facility (PSF). This parameter only applies to IPDS printers which support the Set Media Size (SMS) operation. Refer to Printer Information, S544-5750 for new device support, but the list includes:

- Impact printers: 4224, 4230, 4234, 4247, 6400, 6408, 6412
- Workgroup printers: 3112, 3116, 3812, 3816, 3912, 3916, 3930, 4028, Network printer 12/17/24, Infoprint 20/21/32/40/70/70+/2085/2105
- Thermal printer: 4400

*NO The page size (forms) in the printer is not set by PSF.
*YES The page size (forms) in the printer is set by PSF.

Resident fonts (RESFONT)

Specifies whether Print Services Facility (PSF) supports resident fonts on a printer that has resident fonts. Not supporting resident fonts causes PSF to map the resident font reference to its equivalent host font and then download the host font to the printer.

*YES Resident fonts for the printer are supported by PSF.
*NO Resident fonts for the printer are not supported by PSF. PSF maps the resident font referenced in the spooled file to its equivalent host font and then downloads the host font to the printer.

Resource retention (RSCRET)

Specifies whether resource retention across spooled files is supported by Print Services Facility (PSF).

*YES PSF stores page segments and overlays in the printer across spooled file boundaries. This minimizes data transfers, especially when printing multiple spooled files that reference the same resources.

*NO PSF does not store page segments and overlays in the printer across spooled file boundaries. They are deleted after each spooled file.

Note: The page segments and overlays are deleted in the printer when the printer writer is ended.

Edge orient (EDGEORIENT)

Specifies whether additional page rotation should be performed. When the page rotation value of a spooled file is *COR or *AUTO and the system rotates the output, 90 degree rotation is normally used.

*NO The output remains at its original orientation.
*YES *COR and *AUTO output of 90 degrees is rotated an additional 180 degrees before printing.
Use outline fonts (USEOUTLFNT)
Specifies whether the requested downloadable AFP raster fonts should be replaced with the equivalent downloadable outline fonts.
*NO  The raster fonts will be used.
*YES  If the equivalent downloadable outline font exists, it will be used in place of the raster font.

PSF defined option (PSDFNOPT)
Specifies a value as defined by IBM.
*NONE  No Print Services Facility (PSF) defined options are specified.
character-value  Specify a value as defined by IBM. One or more values may be made available between releases of OS/400. If a value is made available, a PTF cover letter will contain the required syntax.

Font substitution messages (FNTSUBMSG)
Specifies whether Print Services Facility (PSF) will issue messages indicating that a successful font substitution was performed.
*YES  Messages indicating that a successful font substitution was performed are issued.
*NO  Messages indicating that a successful font substitution was performed are not issued. Messages indicating that a font substitution attempt failed will still be issued.

Capture host fonts at printer (FNTCAPTURE)
Specifies whether the printer should capture host downloaded fonts.
*NO  The printer should not capture host fonts.
*YES  The font character set or code page is eligible to be captured after downloading to the printer. If the printer does not support font capturing, this information is ignored and the font is downloaded.

Font resolution for formatting (FNTRSL)
Specifies the resolution Print Services Facility (PSF) should use under these conditions:
• you are printing to a multiple-resolution printer
• the printer is configured to report support of multiple resolutions
• the spooled file does not specify the font metrics and resolution with which to print the spooled file or the font is not available at that resolution
If the printer is configured to report support of either 240 pels per inch or 300 pels per inch only, then PSF will produce the same results as if going to a single resolution printer.

Refer to Printer Device Programming, SC41-5713 for more information regarding the algorithm used for searching a library list for a font resource.

*SEARCH

Search the library list for the first occurrence of a host font with a name match. The resolution of that font will be used to print the spooled file. Message PQT3546 will be issued when this value is selected to indicate to the user the resolution of the font that was finally selected.

- **240** The font resolution used to print the spooled file should be 240 pels per inch.
- **300** The font resolution used to print the spooled file should be 300 pels per inch.

---

**Font mapping table (FNTTBL)**

Specifies the name of a printer-resident to printer-resident font mapping table. Print Services Facility (PSF) uses this font mapping table when printing to a printer which supports printer-resident fonts but the spooled file specifies a printer-resident font that the printer does not support.

For the printer-resident to printer-resident font substitution table, the following processing is done by the system:

- If the printer-resident font specified in the print job is supported by the printer, then it is used. The printer-resident to printer-resident font substitution table is not searched.
- If the printer-resident font specified in the print job is not supported by the printer, then the printer-resident to printer-resident font substitution table is searched.
  - If a matching entry is found in the printer-resident font substitution table and the entry is supported by the printer, then the specified substitute font in the printer-resident font substitution table is used.
  - If a matching entry is not found in the printer-resident font substitution table or if the specified substitute font is not supported by the printer, then the system will use its internal font substitution tables to perform the font substitution.

Refer to Printer Device Programming, SC41-5713 for more information on supported printer-resident fonts. See the CRTFNNTBL, DSPFNNTBL, ADDFNNTBLE, CHGFNNTBLE, and RMVFNNTBLE commands for more information on user font tables.

**Single value**

*NONE

No printer-resident to printer-resident font table is specified. For a print job that references a printer-resident font, if the font is not supported by the printer, the system will substitute another resident font.

**Qualifier 1: Font mapping table**

* name Specify the name of the printer-resident to printer-resident font table.

**Qualifier 2: Library**

* name Specify the name of the library where the font table is located.
Cut sheet emulation mode (CSEMODE)

Specifies to what degree Print Services Facility (PSF) will do size checking of the document when using Cut Sheet Emulation.

*NONE

No checking will be done to verify that the document page will fit on half the continuous forms physical page.

*CHKFIRST

The first page of each copy group will be checked to determine if the page will fit on half the continuous forms page.

*CHKALL

The front side page will be checked to determine if the page will fit on half the continuous forms page.

Use DBCS simulation fonts (MAPIGCFNT)

Specifies to use DBCS simulation fonts instead of the DBCS raster fonts specified in the data stream when printing the spooled file.

DBCS simulation fonts are outline fonts that are positioned like raster fonts. This allows the use of outline fonts to print applications that use DBCS raster fonts without changing the application or the appearance of the printed output. Outline fonts are scalable, so it is not necessary to store font character sets for each point size on your system, and neither is it necessary to download a different font to the printer for every change in point size. This increases your system storage space and enhances printing performance.

*NO Do not substitute DBCS simulation fonts for DBCS raster fonts.

*YES Substitute DBCS simulation fonts for DBCS raster fonts.

Replace (REPLACE)

Specifies whether an existing Print Services Facility (PSF) configuration object with the same name as the one being created, is replaced.

*YES The existing PSF configuration object is replaced.

*NO If a PSF configuration object with same name exists in the library specified, the create operation fails. The existing PSF configuration object is not replaced.

Authority (AUT)

Specifies the authority you are giving to users who do not have specific authority to the object, who are not on an authorization list, and whose group profile has no specific authority to the object.

*LIBCRTAUT

The system determines the authority for the object by using the value specified for the Create authority (CRTAUT) parameter on the Create Library (CRTLIB) command for the library containing the object to be created. If the CRTAUT value for a library is changed by running the Change Library (CHGLIB) command, the new CRTAUT value will not affect any existing objects.
*CHANGE
Change authority allows the user to change and perform basic functions on the object. Change authority provides object operational authority and all data authorities.

*ALL
The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the object's existence, specify the security for the object, change the object, perform basic functions on the object, and change ownership of the object.

*USE
Use authority provides object operational authority, read authority, and execute authority.

*EXCLUDE
The user cannot access the object.

name
Specify the name of an authorization list to be used for authority to the object. Users included in the authorization list are granted authority to the object as specified in the list. The authorization list must exist when the object is created.

Examples
CRTPSFCFG PSFCFG(QGPL/P5001) PDFGEN(*MAIL)
PDFDEVTYPE(*4028) PDFPPRDWR1(*LETTER)
PDFPPRDWR2(*LEGAL) PDFMULT(*YES *INDEX)
PDFDTAQ(*NONE) PDFINCFNT(*YES)
PDFSENDER(QSPLJOB) PDFMAPPGM(*NONE)

This command creates a Print Services Facility (PSF) configuration object named P5001 in the QGPL library. The PDFGEN parameter specifies that the spooled file is to be transformed to PDF and e-mailed by the SNDDST command. The transform is told to emulate a 4028 printer with paper sizes in drawers 1 and 2 to be letter and legal, respectively.

The requested output will have PDF index tags at the group boundaries, the necessary fonts will be placed inline with the output file, the sender is PSF. Since there is no mapping program, the mail tag information associated with the file is assumed to be valid e-mail addresses.

There will be no completion message logged to a data queue because this parameter has a value of *NONE.

Error messages
*ESCAPE Messages
CPF2283
Authorization list &1 does not exist.

CPF88C1
Printer resource type &1 &2 was not created in library &3.

CPF9810
Library &1 not found.

CPF9820
Not authorized to use library &1.

CPF9845
Error occurred while opening file &1.
Create Query Management Form (CRTQMFORM)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Create Query Management Form (CRTQMFORM) command allows you to create a query management form from a specified source file member. The query management form defines how a report is to look when data from running a query is displayed or printed.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>QMFORM</td>
<td>Query management report form</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Query management report form</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>SRCFILE</td>
<td>Source file</td>
<td>Qualified object name</td>
<td>Optional, Positional 2</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Source file</td>
<td>Name, QMFORMSRC</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>SRCMBR</td>
<td>Source member</td>
<td>Name, *QMFORM</td>
<td>Optional, Positional 3</td>
</tr>
<tr>
<td>TEXT</td>
<td>Text ‘description’</td>
<td>Character value, *SRCMBRTEX, *BLANK</td>
<td>Optional</td>
</tr>
<tr>
<td>REPLACE</td>
<td>Replace object</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
</tbody>
</table>

Query management report form (QMFORM)

Specifies the query management form to be created.

This is a required parameter.

This is a required parameter.

Qualifier 1: Query management report form

name Specify the name of the form to be created.

Qualifier 2: Library

*CURLIB

The current library for the job is used to locate the form. If no library is specified as the current library for the job, the QGPL library is used.

name Specify the name of the library where the form is located.
**Source file (SRCFILE)**

Specifies the source file containing the source for the form being created. The form is identified by the name specified on the QMFORM parameter.

Qualifier 1: Source file

QQMFORMSRC

IBM-supplied source file QQMFORMSRC contains the source for the form to be created.

**name** Specify the name of the file containing the source for the form to be created.

Qualifier 2: Library

*LIBL* All libraries in the library list for the current thread are searched until the first match is found.

*CURLIB*

The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

**name** Specify the name of the library to be searched.

---

**Source member (SRCMBR)**

Specifies the source file member containing the form source to be created.

*QMFORM*

The member name is the same as the form name specified on the QMFORM parameter.

**name** Specify the name of the member that contains the form source.

---

**Text ’description’ (TEXT)**

Specifies the text that briefly describes the query management form.

*SRCMBRTXT*

The descriptive text for the object being created is the same as the text description of the source member. If the source member has no text description, or if it is blank, the comment inside the externalized form is used if one exists; otherwise *BLANK is assumed.

*BLANK*

No text is specified. If this command causes an object to be replaced, any existing text for the object is replaced with blank text.

**character-value**

Specify no more than 50 characters of text, enclosed in apostrophes. The apostrophes are not part of the 50-character string.
Authority (AUT)

Specifies the authority you are giving to users who do not have specific authority for the object, who are not on an authorization list, and whose group profile or supplemental group profiles do not have specific authority for the object.

*LIBCRTAUT

The system determines the authority for the object by using the value specified for the Create authority (CRTAUT) parameter on the Create Library command (CRTLIB) for the library containing the object to be created. If the value specified for the CRTAUT parameter is changed, the new value will not affect any existing objects.

*CHANGE

The user can perform all operations on the object except those limited to the owner or controlled by object existence (*OBJEXIST) and object management (*OBJMGT) authorities. The user can change and perform basic functions on the object. *CHANGE authority provides object operational (*OBJOPR) authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove users.

*ALL

The user can perform all operations except those limited to the owner or controlled by authorization list management (*AUTLMGT) authority. The user can control the object’s existence, specify the security for the object, change the object, and perform basic functions on the object. The user also can change ownership of the object.

*USE

The user can perform basic operations on the object, such as running a program or reading a file. The user cannot change the object. Use (*USE) authority provides object operational (*OBJOPR), read (*READ), and execute (*EXECUTE) authorities.

*EXCLUDE

The user cannot access the object.

name Specify the name of an authorization list to be used for authority to the object. Users included in the authorization list are granted authority to the object as specified in the list. The authorization list must exist when the object is created.

Replace object (REPLACE)

Specifies whether an existing object with the same name and type in the library to receive the output is replaced with the output of this command.

*YES An existing object is replaced with the output of this command.

*NO An existing object is not replaced with the output of this command.

Examples

CRTQMFORM QMFORM(FORMEMP) SRCFILE(RPTLIB/FORMSRC)

This command creates a form named FORMEMP in the current library. The form source is in member FORMEMP, which is located in source file FORMSRC in library RPTLIB.
Error messages

*ESCAPE Messages

QWM2701
&1 command failed.

QWM2703
&1 command ended.

QWM2705
Source file &1 in &2 not available.

QWM2706
&1 in &2 not replaced.
Create Query Management Query (CRTQMQRY)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Create Query Management Query (CRTQMQRY) command allows you to create a query management query from a specified source file member. The source for a query is a single Structured Query Language (SQL) statement that can contain variable substitution values and embedded comments. It can be spread over multiple records in a source file member.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>QMQRY</td>
<td>Query management query</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Query management query</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>SRCFILE</td>
<td>Source file</td>
<td>Qualified object name</td>
<td>Optional, Positional 2</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Source file</td>
<td>Name, QMQQRYSRC</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>SRCMBR</td>
<td>Source member</td>
<td>Name, *QMQRY</td>
<td>Optional, Positional 3</td>
</tr>
<tr>
<td>TEXT</td>
<td>Text ‘description’</td>
<td>Character value, *SRCMBRTXT, *BLANK</td>
<td>Optional</td>
</tr>
<tr>
<td>SRTSEQ</td>
<td>Sort sequence</td>
<td>Qualified object name</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>LANGID</td>
<td>Language ID</td>
<td>Character value, *SRC, *JOBRUN, *JOB</td>
<td>Optional</td>
</tr>
<tr>
<td>REPLACE</td>
<td>Replace object</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
</tbody>
</table>

Query management query (QMQRY)

Specifies the query management query to be created.

This is a required parameter.

Qualifier 1: Query management query

name Specify the name of the query to be created.

Qualifier 2: Library
The current library for the job is used to locate the query. If no library is specified as the current library for the job, the QGPL library is used.

**name** Specify the name of the library where the query is located.

Source file (SRCFILE)

Specifies the source file containing the source for the query being created. The query is identified by the name specified on the QMQRY parameter.

**Qualifier 1: Source file**

**QQMQRYSRC** IBM-supplied source file QQMQRYSRC contains the source for the query to be created.

**name** Specify the name of the file containing the source for the query to be created.

**Qualifier 2: Library**

***LIBL** All libraries in the library list for the current thread are searched until the first match is found.

***CURLIB** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

**name** Specify the name of the library to be searched.

Source member (SRCMBR)

Specifies the member of the source file member that contains source for the query to be created.

***QMQRY** The member name is the same as the query name specified on the QMQRY parameter.

**name** Specify the name of the member that contains the query source.

Text 'description' (TEXT)

Specifies the text that briefly describes the query management query.

***SRCMBRTXT** The descriptive text for the object being created is the same as the text description of the source member. If the source member has no text description, or if it is blank, then TEXT(*BLANK) is assumed.

***BLANK** No text is specified. If this command causes an object to be replaced, any existing text for the object is replaced with blank text.

**character-value** Specify no more than 50 characters of text, enclosed in apostrophes. The apostrophes are not part of the 50-character string.
Sort sequence (SRTSEQ)

Specifies the sort sequence table to be used for string comparisons in this query.

**Note:** This parameter is valid only when creating a query. This parameter is not valid when creating a FORM.

**Qualifier 1: Sort sequence**

- **SRC**  The source file member contains the sort sequence to be used when creating the query.
- **JOBRUN**  The SRTSEQ value for the job at the time the query is run is used.
- **JOB**  The SRTSEQ value for the job at the time the query is created is used.
- **HEX**  A sort sequence table is not used, and the hexadecimal values of the characters are used to determine the sort sequence.
- **LANGIDUNQ**  The unique-weight sort table for the language specified on the LANGID parameter is used.
- **LANGIDSHR**  A shared-weight sort table for the language specified on the LANGID parameter is used.

**Qualifier 2: Library**

- **LIBL**  All libraries in the library list for the current thread are searched until the first match is found.
- **CURLIB**  The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

**Language ID (LANGID)**

Specifies the language identifier to be used when SRTSEQ(*LANGIDUNQ) or SRTSEQ(*LANGIDSHR) is specified.

- **SRC**  The source file member contains the language ID to be used when the query is created.
- **JOBRUN**  The LANGID value for the job is determined when the query is run.
- **JOB**  The LANGID value for the job is determined when the query is created.

**language-ID**  Specify the language identifier to be used for the query.
Authority (AUT)

Specifies the authority you are giving to users who do not have specific authority for the object, who are not on an authorization list, and whose group profile or supplemental group profiles do not have specific authority for the object.

*LIBCRTAUT

The system determines the authority for the object by using the value specified for the Create authority (CRTAUT) parameter on the Create Library command (CRTLIB) for the library containing the object to be created. If the value specified for the CRTAUT parameter is changed, the new value will not affect any existing objects.

*CHANGE

The user can perform all operations on the object except those limited to the owner or controlled by object existence (*OBJEXIST) and object management (*OBJMGT) authorities. The user can change and perform basic functions on the object. *CHANGE authority provides object operational (*OBJOPR) authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove users.

*ALL

The user can perform all operations except those limited to the owner or controlled by authorization list management (*AUTLMGT) authority. The user can control the object's existence, specify the security for the object, change the object, and perform basic functions on the object. The user also can change ownership of the object.

*USE

The user can perform basic operations on the object, such as running a program or reading a file. The user cannot change the object. Use (*USE) authority provides object operational (*OBJOPR), read (*READ), and execute (*EXECUTE) authorities.

*EXCLUDE

The user cannot access the object.

name

Specify the name of an authorization list to be used for authority to the object. Users included in the authorization list are granted authority to the object as specified in the list. The authorization list must exist when the object is created.

Replace object (REPLACE)

Specifies whether an object with the same name and type in the library to receive the output is replaced with the output of this command.

*YES

An existing object is replaced with the output of this command.

*NO

An existing object is not replaced with the output of this command.

Examples

CRTMQRY  QMQRY(NEWQRY)  SRCFILE(RPTLIB/QRYSRC)

This command creates a query named NEWQRY in the current library. The query source is in member NEWQRY, which is located in source file QRYSRC in library RPTLIB.
Error messages

*ESCAPE Messages

QWM2701
&1 command failed.

QWM2703
&1 command ended.

QWM2705
Source file &1 in &2 not available.

QWM2706
&1 in &2 not replaced.
Create Q/A Database (CRTQSTDB)

Where allowed to run: Interactive environments (*INTERACT
*IPGM *IREXX *EXEC)
Threadsafe: No

The Create Question-and-Answer Database (CRTQSTDB) command allows you to create a new question-and-answer (Q & A) database. More information is available in the Basic System Operation information in the iSeries Information Center at http://www.ibm.com/eserver/iseries/infocenter.

Restrictions:
1. This command is shipped with public *EXCLUDE authority.
2. The user of this command must have *ADD authority to the library in which the database files are located.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>QSTDB</td>
<td>Q/A database</td>
<td>Name, *SELECT</td>
<td>Optional, Positional 1</td>
</tr>
<tr>
<td>LIB</td>
<td>Lib containing Q/A database</td>
<td>Name, QUSR SYS</td>
<td>Optional, Positional 2</td>
</tr>
</tbody>
</table>

Q/A database (QSTDB)

Specifies the Q & A database to create. When selecting a Q & A database name, the name must be 10 characters or less and must begin with a letter.

The possible values are:

*SELECT
You are asked to specify a Q & A database. If only one Q & A database exists on the system, it is the default.

question-database
Specify the name of the Q & A database in which to create.

Lib containing Q/A database (LIB)

Specifies the name of an existing library that will contain the new Q & A database.

The possible library values are:
QUSRYS

The Q & A database is created in the QUSRYS library.

library-name

Specify name of the library to be searched.

Note: The library must exist on the system.

Examples

CRTQSTDB

This command shows the Create a Q & A Database display.

Error messages

None
Create Q/A Database Load (CRTQSTLOD)

Where allowed to run: Interactive environments (*INTERACT
*IPGM *IREXX *EXEC)
Threadsafe: No

The Create Q & A Database Load (CRTQSTLOD) command allows you to create for distribution a Q & A database load on an alternative medium, such as tape. More information is available in the Basic System Operation information in the iSeries Information Center at http://www.ibm.com/eserver/iseries/infocenter.

Restrictions:
1. This command is shipped with public *EXCLUDE authority.
2. The user must have *READ authority to the Q & A database file referred to by the command.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>QSTDB</td>
<td>Q/A database</td>
<td>Name, *SELECT</td>
<td>Optional, Positional 1</td>
</tr>
<tr>
<td>LIB</td>
<td>Lib containing Q/A database</td>
<td>Name, *QSTLIB</td>
<td>Optional, Positional 2</td>
</tr>
</tbody>
</table>

Q/A database (QSTDB)

Specifies the Q & A database to distribute.

*SELECT

You are asked to specify a Q & A database. If only one Q & A database exists on the system, it is the default.

question-database

Specify the name of the Q & A database that you want to distribute.

Lib containing Q/A database (LIB)

Specifies the name of the library that contains the Q & A database to be used to create the load.

*QSTLIB

The library containing the specified Q & A database is searched. If *SELECT is specified on the QSTDB parameter, any Q & A database in any library to which you are authorized can be selected.
library-name

Specify the name of the library to be searched. If *SELECT is specified on the QSTDB parameter, any Q & A database in the library to which you are authorized can be selected.

Examples

CRTQSTLOD

This command shows the Create a Database Load display.

Error messages

None
Create S/36 Display File (CRTS36DSPF)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Create System/36 Display File (CRTS36DSPF) command creates a display file from a System/36 source file and adds, deletes, or updates formats in existing display files. Using the TOFILE and TOMBR parameters, you can convert the System/36 SFGR source to data description specifications (DDS) source.

Note: The data description specifications (DDS) source is saved in the DDS source file QS36DDSSRC, in the same library as the display file. If the QS36DDSSRC source file does not exist, this source file is created with a record length of 92.

Restriction: Option 5 of the operating system must be installed to run this command. This command can be run either natively or in the System/36 environment.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>DSPFILE</td>
<td>Display file</td>
<td>Qualified object name</td>
<td>Optional, Positional 1</td>
</tr>
<tr>
<td>SRCMBR</td>
<td>S/36 SFGR source member</td>
<td>Single values: *DSPFILE Other values (up to 32 repetitions): Name</td>
<td>Optional, Positional 3</td>
</tr>
<tr>
<td>SRCFILE</td>
<td>S/36 source file</td>
<td>Qualified object name</td>
<td>Optional, Positional 4</td>
</tr>
<tr>
<td>REPLACE</td>
<td>Replace display file</td>
<td>*NO, *YES</td>
<td>Optional</td>
</tr>
<tr>
<td>PRINT</td>
<td>Print SFGR listing</td>
<td>*YES, *NO, *PARTIAL</td>
<td>Optional</td>
</tr>
<tr>
<td>MAXDEV</td>
<td>Maximum devices</td>
<td>1-256, *SRCATR</td>
<td>Optional</td>
</tr>
<tr>
<td>AUT</td>
<td>Authority</td>
<td>Name, *USE, *ALL, *CHANGE, *EXCLUDE, **LIBCRTAUT</td>
<td>Optional</td>
</tr>
<tr>
<td>GENOPT</td>
<td>Generation option</td>
<td>*GEN, *NOGEN, *CONVERT</td>
<td>Optional</td>
</tr>
<tr>
<td>SYNTAX</td>
<td>Check SFGR source syntax</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
<tr>
<td>TOFILE</td>
<td>To DDS source file</td>
<td>Qualified object name</td>
<td>Optional</td>
</tr>
<tr>
<td>TOMBR</td>
<td>To DDS source member</td>
<td>Name, *NONE, *SRCMBR, *DSPFILE</td>
<td>Optional</td>
</tr>
<tr>
<td>HALT</td>
<td>Issue msg if error occurs</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
<tr>
<td>DFRWRT</td>
<td>Defer write</td>
<td>*SRCATR, *YES, *NO</td>
<td>Optional</td>
</tr>
<tr>
<td>TGTRLS</td>
<td>Target Release</td>
<td>Character value, *CURRENT, *PRV</td>
<td>Optional</td>
</tr>
</tbody>
</table>
Display file (DSPFILE)

Specifies the name and the library of the created display file.

This is a required parameter.

display-file-name
   Specify the display file name.

   Note: A display file is not created if the file has the same name and library as an existing program, message file, or other type of file.

The possible library values are:

*CURLIB
   The current library for the job is used to locate the file. If no library is specified as the current library for the job, the QGPL library is used.

library-name
   Specify the name of the library where the display file is located.

Option (OPTION)

Specifies what to do with the specified display file. A maximum of 32 options can be specified. The given option applies to the corresponding name on the S/36 SFGR source member (SRCMBR) parameter.

*CREATE
   A display file is created using the specified display file name as defined by the specified source members. If *CREATE is specified, no other options can be specified.

*ADD
   One or more display formats are added to the specified display file, as defined by the corresponding names specified on the S/36 SFGR source member (SRCMBR) parameter.

*UPDATE
   One or more display formats are updated in the specified display file as defined by the corresponding names specified on the SRCMBR parameter.

*DELETE
   A display format is deleted from the specified display file. The format that is deleted is defined by the corresponding names specified on the SRCMBR parameter. If all formats in the display file are deleted, the entire display file is deleted and a source file member is not produced.

S/36 SFGR source member (SRCMBR)

Specifies the member used in the source file when performing a create, add, or update option. For a delete option specify the name of the format being deleted.

*DSPFILE
   A member with the same name as the display file is specified.

source-member-name
   Specify the source member name. A maximum of 32 source member names can be specified.
S/36 source file (SRCFILE)

Specifies the name of the source file in which the screen format generator (SFG) source members are found.

**QS36SRC**
The source file, QS36SRC, is used.

**file-name**
Specify the name of the source file.

The possible library values are:

**LIBL** All libraries in the library list for the current thread are searched until the first match is found.

**CURLIB** The current library for the job is used to locate the file. If no library is specified as the current library for the job, the QGPL library is used.

**library-name**
Specify the name of the library where the source file is located.

Replace display file (REPLACE)

Specifies whether an existing display file is replaced. When OPTION(*CREATE) is not specified, this parameter is ignored.

**NO** An existing display file is not replaced.

**YES** An existing display file is replaced. Other types of files are not replaced.

**Note:** A display file is not created if the file has the same name and library as a program or message file.

If terminating errors are encountered, the existing display file is not replaced. If the display file already exists, the **Authority (AUT)** parameter is ignored and the authorities for the old display file are copied to the new display file that replaces it. REPLACE(YES) is assumed when OPTION (*ADD) (*UPDATE) or (*DELETE) is specified.

Print SFGR listing (PRINT)

Specifies whether the compile listings are printed.

**YES** Full listings for the SFGR source syntax checking and the DDS compile steps are printed.

**NO** No listing is printed for the SFGR source syntax checking step. *PARTIAL is assumed for the DDS compile step.

**PARTIAL**
If errors are found in any step, the compile listing is printed for that step, along with the error or warning messages. If no messages are issued, no listing is printed.
Maximum devices (MAXDEV)

Specifies the number of devices that can use the display file at one time.

*SRCATR

The maximum number of devices is taken from the first attributes of the source member. The Change System/36 Source Attributes (CHGS36SRCA) and Edit System/36 Source Attributes (EDTS36SRCA) commands can be used to set this value in the source member.

number-of-devices

Specify the maximum number of devices. The valid values range from 1 through 256.

Authority (AUT)

Specifies the authority you are giving the users who do not have specific authority to the object, who are not on the authorization list, and whose user group has no specific authority to the object.

If REPLACE(*YES) is specified and the display file already exists, the AUT parameter is ignored and the authorities for the old display file are copied to the new display file that replaces it. REPLACE(*YES) is assumed when OPTION *ADD, *UPDATE or *DELETE is specified.

*LIBCRTAUT

The authority for the object is the same as the value specified on the Create authority (CRTAUT) parameter of the library in which the object is being created. If the value specified on the CRTAUT parameter is changed, the new value will not affect any existing objects.

*USE

The user can perform basic operations on the object, such as running a program or reading a file. The user cannot change the object. Use (*USE) authority provides object operational (*OBJOPR), read (*READ), and execute (*EXECUTE) authorities.

*ALL

The user can perform all operations except those limited to the owner or controlled by authorization list management (*AUTLMGT) authority. The user can control the object’s existence, specify the security for the object, change the object, and perform basic functions on the object. The user also can change ownership of the object.

*CHANGE

The user can perform all operations on the object except those limited to the owner or controlled by object existence (*OBJEXIST) and object management (*OBJMGT) authorities. The user can change and perform basic functions on the object. *CHANGE authority provides object operational (*OBJOPR) authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove users.

*EXCLUDE

The user cannot access the object.

authorization-list-name

Specify the name of an authorization list to be used for authority to the object. Users included in the authorization list are granted authority to the object as specified in the list. The authorization list must exist when the object is created.
Generation option (GENOPT)

Specifies the compile options that are used.

**GEN**  If *YES* is specified on the Check SFGR source syntax (SYNTAX) parameter, the SFGR source syntax is checked, the display file is converted to data description specifications (DDS), and a display file (*FILE) object is created. If *NO* is specified on the SYNTAX parameter, the SFGR source syntax is not checked.

**NOGEN**  If *YES* is specified on the SYNTAX parameter, the SFGR source syntax is checked.

**CONVERT**  The display file is converted to data description specifications (DDS), but no display file is created. Specify a member name on the To DDS source member (TOMBR) parameter to save the DDS results. If terminating errors are found when syntax checking the SFGR source, no conversion to DDS is performed. If *YES* is specified on the SYNTAX parameter, SFGR syntax checking is performed.

Check SFGR source syntax (SYNTAX)

Specifies whether the SFGR source syntax is checked.

**YES**  The source syntax is checked.

**NO**  The source syntax is not checked.

To DDS source file (TOFILE)

Specifies the name of the source file in which to store the DDS source that is used to create the display file. The file need not already exist. If the user is authorized to the CRTSRCPF (Create Source Physical File) command, and the file does not exist, a new source file is created.

**QDDSSRC**  The source file, QDDSSRC, is used.

Specify the name of the source file. The file should have a record length of 92 bytes.

The possible library values are:

**LIBL**  All libraries in the library list for the current thread are searched until the first match is found.

**CURLIB**  The current library for the job is used to locate the file. If no library is specified as the current library for the job, the QGPL library is used.

Specify the name of the library where the source file is located.
To DDS source member (TOMBR)

Specifies the name of the source file member in which to store the DDS source. The source file member is added if it does not exist, and is replaced if it exists. When the member name is the same as that of the display file name, and the to-file is QS36DDSSRC in the same library as the display file being created, the DDS is saved in this member only if the compile operation of the display file is successful. To guarantee that the DDS is saved, specify the name of some other source file, library, or member.

*NONE

The DDS source is not stored in the source file specified on the To DDS source file (TOFILE) parameter. The source is saved in the Q36DDSSRC source file.

*SRCMBR

The first name specified on the S/36 SFGR source member (SRCMBR) parameter is used as the member name.

*DSPFILE

The display file name is used as the member name.

member-name

Specify the name of the source file member in which to store the DDS source. If the member does not exist, it is added. If it already exists, it is replaced. If any terminating errors are encountered, the member is not added or changed.

Issue msg if error occurs (HALT)

Specifies whether an error message is issued when an SFGR syntax error is detected.

*YES The request ends with an error message.

*NO The request ends with a diagnostic message and a return code of 1008 is set.

Defer write (DFRWRT)

Specifies that data is not written to the display file until a read request is made. Control is returned to the requesting program immediately after the data is received for output. This may result in improved performance.

*SRCATR

The defer write option is taken from the DFRWRT attribute of the first source member. If there is no source member, or the DFRWRT source attribute has not been set, *YES is used.

*YES When the write request is made to the display file, control is returned after the buffer is processed. The actual display of the data may take place after a read or combined write/read operation is performed. The program buffer is immediately available for the next read or combined write/read operation.

*NO When the write request is made to the display file, control is not returned to the requesting program until the input/output request is completed including displaying the data and making input/output information available.
**Target Release (TGTRLS)**

Specifies the release level of the operating system on which you intend to use the object being saved.

When specifying the **target-release** value, the format VxRxMx is used to specify the release, where Vx is the version, Rx is the release, and Mx is the modification level. For example, V5R3M0 is version 5, release 3, modification 0.

Valid values depend on the current version, release, and modification level of the operating system, and they change with each new release. You can press F4 while prompting this command parameter to see a list of valid target release values.

**CURRENT**

The object is to be restored to, and used on, the release of the operating system currently running on your system. The object can also be restored to a system with any subsequent release of the operating system installed.

**PRV**

The object is to be restored to the previous release with modification level 0 of the operating system. The object can also be restored to a system with any subsequent release of the operating system installed.

character-value

Specify the release in the format VxRxMx. The object can be restored to a system with the specified release or with any subsequent release of the operating system installed.

**Examples**

CRTS36DSPF  DSPFILE(MYLIB/MYDISPLAY)  OPTION(*CREATE)
SRCMBR(SFGRMBR1 SFGRMBR2)
SRCFILE(MYLIB/QS36SRC)  GENOPT(*NOGEN)

This command checks the SFGR source (producing a printer list named QPUTSFGR). No DDS is built, and the display file is not created.

**Error messages**

**ESCAPE Messages**

**SSP4464**

Member &3 in file &1 in use, cannot be shared.

**SSP5003**

Source member &1 not found.

**SSP5004**

&1—This load member exists, but is not a $SFGR member.

**SSP5005**

&1 display file already exists.

**SSP5007**

Source member &1 already given.

**SSP5009**

Display file &1 not found in library &2.
SSP5010
&1 not System/36 display file.

SSP5011
&1 not allowed for display file name.

SSP5012
Format &1 not found in display file

SSP5015
Source file library &1 not found.

SSP5016
Display file library &1 not found.

SSP5017
TOFILE library &1 not found.

SSP5019
Terminating errors in $SFGR input specifications.

SSP5027
TGTRLS(*PRV) allowed with changes only when existing display file created for previous release.

SSP5451
Existing file &1 is not a display file.

SSP6124
Unexpected error occurred.

SSP7375
Error &1 received by &2 utility.

SSP8663
User not authorized to access &1.

SSP8679
Not authorized to access member &1.

SSP9080
Object &1 in use; it cannot be shared.
Create S/36 Menu (CRTS36MNU)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Create System/36 Menu (CRTS36MNU) command creates a menu (display file and command message file) from your source members. You can specify that this menu be created either in a fixed-format, with options 1 through 24 arranged in two columns, or in free-format.

Restriction: Option 5 of the operating system must be installed to run this command. This command can be run either natively or in the System/36 environment.

| Parameters |

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMDTXTMBR</td>
<td>Command text source member##</td>
<td>Name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td>OPTTXTMBR</td>
<td>Option text source member</td>
<td>Name, *NONE</td>
<td>Optional, Positional 2</td>
</tr>
<tr>
<td>CMDTXTSRC</td>
<td>Command text source file</td>
<td>Qualified object name</td>
<td>Optional, Positional 3</td>
</tr>
<tr>
<td>OPTTXTSRC</td>
<td>Option text source file</td>
<td>Qualified object name</td>
<td>Optional, Positional 4</td>
</tr>
<tr>
<td>MNULIB</td>
<td>Menu library (LOADLIB)</td>
<td>Name, *CMDLIB, *CURLIB</td>
<td>Optional, Positional 5</td>
</tr>
<tr>
<td>REPLACE</td>
<td>Replace menu</td>
<td>*NO, *YES</td>
<td>Optional</td>
</tr>
<tr>
<td>FREEFORM</td>
<td>Free form menu</td>
<td>*NO, *YES</td>
<td>Optional</td>
</tr>
<tr>
<td>KEEP</td>
<td>Keep option text msg file</td>
<td>*NO, *YES</td>
<td>Optional</td>
</tr>
<tr>
<td>DDSLIST</td>
<td>DDS listing</td>
<td>*PARTIAL, *FULL</td>
<td>Optional</td>
</tr>
<tr>
<td>MAXDEV</td>
<td>Maximum devices</td>
<td>1-256, 5</td>
<td>Optional</td>
</tr>
<tr>
<td>TOFILE</td>
<td>To DDS source file</td>
<td>Qualified object name</td>
<td>Optional</td>
</tr>
<tr>
<td>TOMBR</td>
<td>To DDS source member</td>
<td>Name, *NONE</td>
<td>Optional</td>
</tr>
<tr>
<td>IGCDTA</td>
<td>User specified DBCS data</td>
<td>*NO, *YES</td>
<td>Optional</td>
</tr>
<tr>
<td>TGTRLS</td>
<td>Target Release</td>
<td>Character value, *CURRENT, *PRV</td>
<td>Optional</td>
</tr>
</tbody>
</table>
Command text source member## (CMDTXTMBR)

Specifies the source member that contains the System/36 message source that is used to create the command text message file. This message file defines the commands that are used when an option is selected. The message IDs must begin with USR.

This is a required parameter.

member-name##

Specify the name of the source file member to use for the command text message file being created. The trailing ## symbols are required. The menu (display file) name is the same as the member name without the ## symbols.

Note: The message file name specified within the source member must be the same as the source member name.

Option text source file (OPTTXTMBR)

Specifies the source member that is used to create the option text message file, or a screen format generator (SFGR) source member used to create the menu display file.

*NONE

The option text is taken from the command text message file. If OPTTXTMBR(*NONE) is specified, FREEFORM(*NO) must also be specified.

member-name

Specify the name of the source member to use for the descriptions of the options on the menu you are creating. The member can contain either the SFGR or the System/36 message source. A source member must be specified when *YES is specified on the Free form menu (FREEFORM) parameter.

Command text source file (CMDTXTSRC)

Specifies the source file in which the command text member is located.

QS36SRC

The source file, QS36SRC, is used.

file-name

Specify the name of the source file that contains the member specified on the Command text source member## (CMDTXTMBR) parameter.

The possible library values are:

*CURLIB

The current library for the job is used to locate the file. If no library is specified as the current library for the job, the QGPL library is used.

library-name

Specify the name of the library where the file is located.
Option text source file (OPTTXTSRC)

Specifies the name and library of the source file in which the option text member is located.

**QS36SRC**

The source file, QS36SRC, is used.

**file-name**

Specify the name of the source file that contains the member specified on the **Option text source member (OPTTXTMBR)** parameter.

The possible library values are:

* **CMDLIB**
  
  The library specified on the **Command text source file (CMDTXTSRC)** parameter is used to locate the file.

* **CURLIB**
  
  The current library for the job is used to locate the file. If no library is specified as the current library for the job, the QGPL library is used.

**library-name**

Specify the name of the library where the file is located.

---

Menu library (LOADLIB) (MNULIB)

Specifies the menu library that is used to store the created menu.

* **CMDLIB**
  
  The library specified on the **Command text source file (CMDTXTSRC)** parameter is used to store the file.

* **CURLIB**
  
  The current library for the job is used to store the file. If no library is specified as the current library for the job, the QGPL library is used.

**library-name**

Specify the name of the library where the menu is located.

---

Replace menu (REPLACE)

Specifies whether an existing display file is replaced.

* **NO** An existing display file is not replaced.

* **YES** An existing display file is replaced by the one being created.

  **Note:** The menu is not created if it has the same name and library as an existing program or message file.

If terminating errors are encountered, the existing display file is not replaced. If the display file already exists, the AUT parameter is ignored, and the authorities for the old display file are copied to the new display file that replaces it.
Free form menu (FREEFORM)

Specifies whether the menu is created in free-format or in a fixed-format.

*NO  Free-format is not used. The menu is created using a fixed-format with two-columns. The message number corresponds to the option number.

*YES  The menu is created using free-format. If *YES is specified, a value must be specified on the Option text source member (OPTTXTMBR) parameter. Option text message numbers correspond to the row numbers on the screen.

Keep option text msg file (KEEP)

Specifies whether the option text message file is kept when the compilation is complete.

*NO  The option text message file is not kept.

*YES  The option text message file is kept. If the option text source member contains the screen format generator (SFGR) source and has the same name as the menu, then KEEP(*YES) is required.

DDS listing (DDSLIST)

Specifies whether a partial or full DDS compile listing is provided.

*PARTIAL  A partial listing is provided.

*FULL  A full DDS listing and cross-reference are provided.

Maximum devices (MAXDEV)

Specifies the maximum number of devices that can use the menu at one time.

5  The maximum number of devices is five.

number-of-devices  Specify the maximum number of devices. Valid values range from 1 through 256.

Authority (AUT)

Specifies the authority you are giving the users who do not have specific authority to the object, who are not on the authorization list, and whose user group has no specific authority to the object.

If REPLACE(*YES) is specified and the display file already exists, the Authority (AUT) parameter is ignored and the authorities for the old display file are copied to the new display file that replaces it.

*LIBCRTAUT  The authority for the object is the same as the value specified on the Create authority (CRTAUT) parameter of the library in which the object is being created. If the value specified on the CRTAUT parameter is changed, the new value will not affect any existing objects.
*USE The user can perform basic operations on the object, such as running a program or reading a file. The user cannot change the object. Use (*USE) authority provides object operational (*OBJOPR), read (*READ), and execute (*EXECUTE) authorities.

*ALL The user can perform all operations except those limited to the owner or controlled by authorization list management (*AUTLMGT) authority. The user can control the object’s existence, specify the security for the object, change the object, and perform basic functions on the object. The user also can change ownership of the object.

*CHANGE The user can perform all operations on the object except those limited to the owner or controlled by object existence (*OBJEXIST) and object management (*OBJMGT) authorities. The user can change and perform basic functions on the object. *CHANGE authority provides object operational (*OBJOPR) authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove users.

*EXCLUDE The user cannot access the object.

authorization-list-name Specify the name of an authorization list to be used for authority to the object. Users included in the authorization list are granted authority to the object as specified in the list. The authorization list must exist when the object is created.

To DDS source file (TOFILE) Specifies the name of the source file in which to store the DDS source that is used to create the menu. The file need not already exist. If you are authorized to the CRTSRCPF (Create Source Physical File) command, and the file does not exist, a new source file is created.

QDDSSRC The source file, QDDSSRC, is used.

file-name Specify the name of the source file in which to store the DDS source.

The possible library values are:

*MNULIB The menu library is used to locate the source file.

*CMDLIB The command library is used to locate the source file.

*CURLIB The current library for the job is used to locate the source file. If no library is specified as the current library for the job, the QGPL library is used.

library-name Specify the name of the library where the source file is located.
To DDS source member (TOMBR)

Specifies the name of the source file member in which to store the DDS source. When the member name is the same as that of the display file name, and the to-file is QS36DDSSRC in the same library as the display file being created, the DDS is saved in this member only if the compile operation of the display file is successful. To guarantee that the DDS is saved, specify the name of some other source file, library, or member.

*NONE
The DDS source is not stored in the source file specified on the To DDS source file (TOFILE) parameter.

member-name
Specify the name of the source file member in which to store the DDS source. If the member does not exist, it is added. If it already exists, it is replaced.

User specified DBCS data (IGCDTA)

Specifies whether the display file contains double-byte character data.

*NO
The display file contains no double-byte character data. Option text message IDs must begin with USR.

*YES
The display file or the message files contain double-byte character data. Option text message IDs may begin with USZ. To get the full benefit of this function, an IGC version of the operating system must be installed.

Target Release (TGTRLS)

Specifies the release level of the operating system on which you intend to use the object being saved.

When specifying the target-release value, the format VxRxMx is used to specify the release, where Vx is the version, Rx is the release, and Mx is the modification level. For example, V5R3M0 is version 5, release 3, modification 0.

Valid values depend on the current version, release, and modification level of the operating system, and they change with each new release. You can press F4 while prompting this command parameter to see a list of valid target release values.

*CURRENT
The object is to be restored to, and used on, the release of the operating system currently running on your system. The object can also be restored to a system with any subsequent release of the operating system installed.

*PRV
The object is to be restored to the previous release with modification level 0 of the operating system. The object can also be restored to a system with any subsequent release of the operating system installed.

character-value
Specify the release in the format VxRxMx. The object can be restored to a system with the specified release or with any subsequent release of the operating system installed.
This command creates a free format menu display from the specified SFGR source member, placing the resulting menu into library WORKLIB. Since the SFGR source is being used, FREEFORM(*YES) and KEEP(*YES) are required. The layout of the menu display is controlled by the format definition located in the SFGR source member. No special checking is done to ensure that the display file defined by this source is valid when used by the system as a menu.
SSP5761
  Option text message file has wrong name.

SSP5763
  Option text member name should not be same as menu name.

SSP5766
  Option text source library &1 was not found.

SSP5767
  Menu library &1 not found.

SSP5768
  Command text source library &1 not found.

SSP5772
  Option text source member required for free format menu.

SSP5773
  Command and option text member names must not be the same.

SSP6124
  Unexpected error occurred.

SSP7375
  Error &1 received by &2 utility.

SSP8663
  User not authorized to access &1.

SSP8679
  Not authorized to access member &1.

SSP9080
  Object &1 in use; it cannot be shared.
Create S/36 Message File (CRTS36MSGF)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Create System/36 Message File (CRTS36MSGF) command creates a message file from your System/36 message source member. This enables you to convert your System/36 message source to the message source on this system.

System/36 message source consists of 3 types of statements:

1. The message control statement must be the first record in the source. Only one control statement is allowed. The control statement has the following syntax:
   
   \[ \text{name,level} \text{<comment>} \]
   
   where: **name** is the name of the message file being created or changed. The name must start in column 1 of the source record. Level must be a 1 or 2 or a blank, and is separated from the name by a comma. A level 1 indicates that the source is for message text (first-level messages); a level 2 indicates that the source is for online help information (second-level messages). If level is omitted or blank, a first level message is assumed. Anything after the first blank in the control statement is considered to be a comment and is ignored.

2. Comment records that start with an asterisk (* in column 1. All comment records are ignored.

3. One or more message text statements may be defined. The message text statement has the following syntax:

   \[ \text{mmmm text} \]

   where: **mmmm** is the System/36 Message Identification Code (MIC) and must start in column 1 and consist of 4 digits (0-9). All 4 digits are required. The 7-character message identifier is created by adding the message prefix specified by the MSGPFX parameter in front of the MIC. Column 5 is ignored and should be left blank. Text is the text of the message being defined and starts in column 6. The message text statements must be arranged so that the MICs are not in descending order. If there is insufficient room to define the entire message in a single record, the MIC can be repeated in column 1 of the next record and the text can be continued starting in column 6.

   A first level message is restricted to a maximum of 75 characters, and a second level message is restricted to a maximum of 225 characters unless RESTRICT(*NO) is specified. Trailing blanks on the last record for each MIC are not counted. The record length used when processing the SRCMBR is the RCDLEN source attribute for that member. This attribute defaults to the source file record length, but can be set or changed when using the CHGS36SRCA, EDTS36SRCA, or RSTS36LIBM commands. The RCDLEN source attribute is the member’s logical record length. If the RCDLEN source attribute is less than the record length of the source file, all characters after the logical record length are ignored.

   If any record contains any non-blank characters after the logical record length, a diagnostic message is issued as a warning.

**Restriction:** Option 5 of the operating system must be installed to run this command. This command can be run either natively or in the System/36 environment.
### Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>SRCMBR</td>
<td>S/36 message source member</td>
<td>Name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td>SRCFILE</td>
<td>S/36 source file</td>
<td>Qualified object name</td>
<td>Optional, Positional 2</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: S/36 source file</td>
<td>Name, QS36SRC</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>MSGLIB</td>
<td>Message file library</td>
<td>Name, *CURLIB</td>
<td>Optional, Positional 3</td>
</tr>
<tr>
<td>REPLACE</td>
<td>Replace message file</td>
<td>*NO, *YES</td>
<td>Optional</td>
</tr>
<tr>
<td>MSGPFX</td>
<td>Message identifier prefix</td>
<td>Name, USR</td>
<td>Optional</td>
</tr>
<tr>
<td>OPTION</td>
<td>Option</td>
<td>*CREATE, *ADD, *CHANGE</td>
<td>Optional</td>
</tr>
<tr>
<td>SUBST</td>
<td>Allow # substitution fields</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
<tr>
<td>RESTRCT</td>
<td>Enforce S/36 restrictions</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
<tr>
<td>TOFILE</td>
<td>To CL source file</td>
<td>Qualified object name</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: To CL source file</td>
<td>Name, QCIRSRC, *NONE</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>TOMBR</td>
<td>To CL source member</td>
<td>Name, *NONE, *SRCMBR</td>
<td>Optional</td>
</tr>
<tr>
<td>HALT</td>
<td>Issue msg if error occurs</td>
<td>*YES, *NO, *IGNORE</td>
<td>Optional</td>
</tr>
</tbody>
</table>

### S/36 message source member (SRCMBR)

Specifies the source member that is used to create the message file. The first record that is not a comment record in the source member specifies the name of the message file being created or changed. The first record that is not a comment record also specifies whether the text in this source member is for the first-level or second-level message text.

This is a required parameter.

### S/36 source file (SRCFILE)

Specifies the source file that is used to create the message file.

**QS36SRC**

The source file QS36SRC is used.

*file-name*

Specify the name of the source file that contains the member specified on the **S/36 message source member (SRCMBR)** parameter.

The possible library values are:

*LIBL  All libraries in the library list for the current thread are searched until the first match is found.
*CURLIB The current library for the job is used to locate the file. If no library is specified as the current library for the job, the QGPL library is used.*
library-name
   Specify the name of the library where the source file is located.

Message file library (MSGLIB)
Specifies the library that is used to store the message file being created.

*CURLIB
   The current library for the job is used to store the message file. If no library is specified as the current library for the job, the QGPL library is used.

*SRCLIB
   The source library is used to store the message file.

library-name
   Specify the name of the library where the created message file is stored.

Replace message file (REPLACE)
Specifies whether or not an existing message file is replaced. This parameter is ignored if OPTION(*CREATE) is not specified.

*NO
   An existing message file is not replaced.

*YES
   An existing file is replaced by the one being created.

   Note: The message file is not created if it has the same name and library as an existing program or display file.

   If any terminating errors are encountered, the existing message file is not replaced. If the message file already exists, the Authority (AUT) parameter is ignored and all of the authorities for the old message file are copied to the new message file that replaces it.

Message identifier prefix (MSGPFX)
Specifies the message prefix that is used to create messages. The default prefix of USR must be used if the message is used to build a menu.

USR
   The message prefix USR is used.

message-prefix
   Specify the three-character message prefix to use with message identifiers.

Option (OPTION)
Specifies what to do with the message file that is specified in the source member.

*CREATE
   A message file is created with the specified name.

*ADD
   Messages are added to the specified message file.
*UPDATE

Messages are changed in the specified message file. If a message does not exist, it is created and added to the message file.

Allow # substitution fields (SUBST)

Specifies that System/36 text replacement fields are converted into the notation that represents text replacement fields in message file (*MSGF) objects. Although *YES is the default, not everyone uses text replacement fields and *NO is a way to make sure that a field is not converted by mistake. Replacement fields are represented by a pair of trailing ## symbols in the S/36 source.

*YES  The replacement text fields are converted. A string of # symbols in the message text source is converted to a replacement text field only when the string occurs at the start or end of the message text, or when the string is preceded and followed by one of the following delimiter characters:

'    blank
.    period
<    less than
(    left parenthesis
+    plus
&    ampersand
*    asterisk
)    right parenthesis
;    semicolon
-    minus
,    comma
>    greater than
?    question
:    colon
'    apostrophe
=    equal
"    double quote

*NO  The replacement text fields are not converted.

Enforce S/36 restrictions (RESTRICT)

Specifies whether System/36 restrictions on message text length are enforced.

*YES  System/36 restrictions are enforced. Message text is limited to 75 characters. Online help for messages is limited to 225 characters.

*NO  System/36 restrictions are not enforced.
Authority (AUT)

Specifies the authority you are giving to users who do not have specific authority to the object, who are not on the authorization list, and whose users’ group has no specific authority to the object.

If OPTION(*CREATE) and REPLACE(*YES) is specified and the message file already exists, the Authority (AUT) parameter is ignored and all of the authorities for the old message file are copied to the new message file that replaces it. The AUT parameter is always ignored when OPTION (*ADD) or (*UPDATE) is specified.

*LIBCRTAUT
The authority for the object is the same as the value specified on the Create authority (CRTAUT) parameter of the library in which the object is being created. If the value specified on the CRTAUT parameter is changed, the new value will not affect any existing objects.

*USE
The user can perform basic operations on the object, such as running a program or reading a file. The user cannot change the object. Use (*USE) authority provides object operational (*OBJOPR), read (*READ), and execute (*EXECUTE) authorities.

*ALL
The user can perform all operations except those limited to the owner or controlled by authority list management (*AUTLMGT) authority. The user can control the object’s existence, specify the security for the object, change the object, and perform basic functions on the object. The user also can change ownership of the object.

*CHANGE
The user can perform all operations on the object except those limited to the owner or controlled by object existence (*OBJEXIST) and object management (*OBJMGT) authorities. The user can change and perform basic functions on the object. *CHANGE authority provides object operational (*OBJOPR) authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove users.

*EXCLUDE
The user cannot access the object.

authorization-list-name
Specify the name of an authorization list to be used for authority to the object. Users included in the authorization list are granted authority to the object as specified in the list. The authorization list must exist when the object is created.

To CL source file (TOFILE)

Specifies the name and library of the source file in which to store the CL source that was used to create the message file. If the file does not already exist, a new source file is created if the user is authorized to the CRTSRCCPF (Create Source Physical File) command.

QCLSRC
The source file, QCLSRC, is used.

file-name
Specify the name of the source file in which the source is stored.
The possible library values are:

*CURLIB

The current library for the job is used to locate the file. If no library is specified as the current library for the job, the QGPL library is used.

library-name

Specify the name of the library where the source file is located.

---

To CL source member (TOMBR)

Specifies the name of the source file member in which to store the data description specifications (DDS) source. If the member does not exist, it is created. When the member name is the same as that of the display file name, and the to-file is QS36DDSSRC in the same library as the display file being created, the DDS is saved in this member only if the compile operation of the display file is successful. To guarantee that the DDS is saved, specify the name of some other source file, library, or member.

*NONE

The CL source is not stored in the source file specified on the To CL source file (TOFILE) parameter.

*SRCMBR

If no member name is specified, the member name specified on the S/36 message source member (SRCMBR) parameter is used. The same name and library should not be specified for both the TOFILE parameter and the SRCFILE parameter, since this would destroy your original System/36 message source.

member-name

Specify the name of the source file member in which to store the CL source. The specified member is replaced if it exists. If it does not exist, it is created. If any terminating errors are encountered, the member is not added or changed.

---

Issue msg if error occurs (HALT)

Specifies whether processing stops when an error is detected.

*YES

The request ends with an error message.

*NO

The request ends with a diagnostic message, and a return code of 2034 is set.

*IGNORE

If an error that can be ignored is detected in the source, a diagnostic message is issued, and the current source statement is ignored. Processing continues with the next source statement. If the error cannot be ignored, processing stops and an error message is issued.

---

Examples

```
CRTS36MSGF SRCMBR(MYMSGSRC) SRCFILE(MYLIB/QS36SRC)
MSGLIB(WORKLIB) REPLACE(*YES)
```

This command creates a message file named MYMSGSRC in library WORKLIB. If a message file with the same name already exists, this new message file replaces the existing one.
**Error messages**

*ESCAPE Messages*

**SSP1727**
Message file &1 not found in library &2.

**SSP5017**
TOFILE library &1 not found.

**SSP6124**
Unexpected error occurred.

**SSP7375**
Error &1 received by &2 utility.

**SSP8663**
User not authorized to access &1.
Create Save File (CRTSAVF)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Create Save File (CRTSAVF) command creates a save file.

A save file can be used with save and restore commands to contain data that would otherwise be written to tape or optical media. A save file can also be used like a database file to read or write records that contain save or restore information. A save file can also be used to send objects to another user on the Systems Network Architecture distribution services (SNADS) network.

Restrictions:
- An online save file should only be used for save/restore data. When restoring objects from the save file, the data in the file must have been produced by a save command. When records are written to a save file by using a high-level language program, the records must contain data produced by saving objects into the save file.
- The records retrieved from a save file contain sequencing checksum information that is validated by the system when records are inserted into a save file. An attempt to insert records that are either out of sequence or have been changed since retrieval from a save file are rejected.
- A save file can contain the output of only one save operation and one library, and all objects in the save file must have been saved from the same library. If the Send Network File (SNDNETF) command is used to send a save file, the maximum size of the save file is approximately 2 billion bytes. The number of objects saved in a save file is limited to the number of objects saved when saving to tape or optical media, as long as the file’s maximum size is not exceeded. The exact number of objects saved into a save file depends on the object types and the object contents.
- You must have object operational (*OBJOPR) authority to the CRTSAVF command in order to create a duplicate save file object or to restore a save file object.
- You must have add (*ADD) and read (*READ) authority to the library in which the save file is to be created.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>FILE</td>
<td>Save file</td>
<td>Qualified object name</td>
<td>Required,</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Save file</td>
<td>Name</td>
<td>Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>TEXT</td>
<td>Text ‘description’</td>
<td>Character value, *BLANK</td>
<td>Optional</td>
</tr>
<tr>
<td>MAXRCDS</td>
<td>Maximum records</td>
<td>1-2146762800, *NOMAX</td>
<td>Optional</td>
</tr>
<tr>
<td>ASP</td>
<td>ASP number</td>
<td>1-32, *LIBASP</td>
<td>Optional</td>
</tr>
<tr>
<td>WAITFILE</td>
<td>Maximum file wait time</td>
<td>Integer, *IMMED, *CLS</td>
<td>Optional</td>
</tr>
<tr>
<td>SHARE</td>
<td>Share open data path</td>
<td>*NO, *YES</td>
<td>Optional</td>
</tr>
</tbody>
</table>
Save file (FILE)
Specifies the save file to be created.

If the file is used in a high-level language program, the file name must be consistent with the naming rules of that language. Otherwise, the file must be renamed in the program.

This is a required parameter.

Qualifier 1: Save file
name Specify the name of the save file to be created.

Qualifier 2: Library
*CURLIB
The current library for the job is used to locate the save file. If no library is specified as the current library for the job, QGPL is used.
name Specify the library where the save file is located.

Text 'description' (TEXT)
Specifies the text that briefly describes the object.

*BLANK
No text is specified.

character-value
Specify no more than 50 characters of text, enclosed in apostrophes.

Maximum records (MAXRCDS)
Specifies the maximum number of records the save file can contain. The number of bytes of space in the save file is estimated at 8192 + (512 x the number of records in the save file). There is room for approximately two thousand 512-byte records in 1 megabyte of space. If you wanted to ensure that the save file would not exceed approximately 20 megabytes you would specify 40000 records (20 megabytes x 2000 records/megabyte).

Note: The maximum amount of data that a save file can contain is approximately 1 terabyte. A message appears when the file is full.

*NOMAX
The maximum value of 2146762800 records is used.

1-2146762800
Specify the maximum number of records the save file can contain.

Auxiliary storage pool ID (ASP)
Specifies the auxiliary storage pool (ASP) from which the system allocates storage for the save file.
The storage space for the save file is allocated from the same auxiliary storage pool that the save file’s library is allocated from.

Specify the identifier of the auxiliary storage pool.

**Maximum file wait time (WAITFILE)**

Specifies the number of seconds that the program waits for the file resources to be allocated when the file is opened, or the device or session resources to be allocated when an acquire operation is performed to the file. If the file resources cannot be allocated in the specified wait time, an error message is sent to the program.

**IMMED**
The program does not wait. Immediate allocation of file resources is required.

**CLS**
The job default wait time is used as the wait time for the file resources to be allocated.

Specify the number of seconds to wait for file resources to be allocated.

**Share open data path (SHARE)**

Specifies whether the open data path (ODP) is shared with other programs in the same routing step. When an ODP is shared, the programs accessing the file share facilities such as the file status and the buffer.

**NO**
The ODP is not shared with other programs in the routing step. A new ODP for the file is created and used every time a program opens the file.

**YES**
The same ODP is shared with each program in the job that also specifies *YES when it opens the file.

**Authority (AUT)**

Specifies the authority you are giving to users who do not have specific authority for the object, who are not on an authorization list, and whose group profile or supplemental group profiles do not have specific authority for the object.

**EXCLUDE**
The user cannot access the object.

**ALL**
The user can perform all operations except those limited to the owner or controlled by authorization list management (*AUTLMGT) authority. The user can control the object’s existence, specify the security for the object, change the object, and perform basic functions on the object. The user also can change ownership of the object.

**CHANGE**
The user can perform all operations on the object except those limited to the owner or controlled by object existence (*OBJEXIST) and object management (*OBJMGT) authorities. The user can change and perform basic functions on the object. *CHANGE authority provides object operational (*OBJOPR) authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove users.
*LIBCRTAUT
The system determines the authority for the object by using the value specified for the Create authority (CRTAUT) parameter on the Create Library command (CRTLIB) for the library containing the object to be created. If the value specified for the CRTAUT parameter is changed, the new value will not affect any existing objects.

*USE The user can perform basic operations on the object, such as running a program or reading a file. The user cannot change the object. Use (*USE) authority provides object operational (*OBJOPR), read (*READ), and execute (*EXECUTE) authorities.

name Specify the name of an authorization list. Users included on the authorization list are granted authority to the object as specified by the list. The authorization list must exist when the object is created.

Examples
CRTSAVF  FILE(ONLINE)  TEXT('Online save file')

This command creates an online save file named ONLINE in the current library. The save file is in the system ASP with no maximum number of records. The public has no authority to this file; only the object owner and users that have the object owner user profile as their group profile can use this save file.

Error messages
*ESCAPE Messages
CPF7302
File &1 not created in library &2.
Create Subsystem Description (CRTSBSD)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Create Subsystem Description (CRTSBSD) command creates a subsystem description that defines the operational attributes of a subsystem. After the subsystem description is created, it can be specialized by commands that add, change, and remove work entries and routing entries in the subsystem description.

Restrictions:
1. To use this command, you must have:
   • read (*READ) and add (*ADD) authority to the library where the subsystem description is to be created.
   • all object (*ALLOBJ) and security administration (*SECADM) special authority to specify a value other than *NONE for a system library list entry.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>SBSD</td>
<td>Subsystem description</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td>Qualifier 1: Subsystem description</td>
<td>Name</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qualifier 2: Library</td>
<td>Name, *CURLIB</td>
<td></td>
<td></td>
</tr>
<tr>
<td>POOLS</td>
<td>Storage pools</td>
<td>Values (up to 10 repetitions): Element list</td>
<td>Required, Positional 2</td>
</tr>
<tr>
<td>Element 1: Pool identifier</td>
<td>1-10</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Element 3: Activity level</td>
<td>Integer</td>
<td></td>
<td></td>
</tr>
<tr>
<td>MAXJOBS</td>
<td>Maximum jobs</td>
<td>0-1000, *NOMAX</td>
<td>Optional, Positional 3</td>
</tr>
<tr>
<td>TEXT</td>
<td>Text ‘description’</td>
<td>Character value, *BLANK</td>
<td>Optional</td>
</tr>
</tbody>
</table>
### Subsystem description (SBSD)

Specifies the name and library of the subsystem description being created. The subsystem description is stored in the specified library.

This is a required parameter.

**Qualifier 1: Subsystem description**

*name*  Specify the name of the subsystem description being created.

**Qualifier 2: Library**

* CURLIB  
  The current library of the thread is used. If no current library exists for the thread, library QGPL is used.

*name*  Specify the library where the subsystem description will be created.

For more information on subsystem descriptions, see the Work Management Guide.

### Storage pools (POOLS)

Specifies one or more storage pool definitions that are in this subsystem description. Each definition specifies for one storage pool:

- Pool definition identifier: The identifier inside the subsystem description, of the storage pool definition. The same identifiers (1 through 10) can be used for pool definitions in different subsystem descriptions.
- Size: The size of the storage pool, expressed in kilobyte (1K = 1024 bytes) multiples. This is the amount of main storage that can be used by the pool.
- Activity level: The maximum number of threads that can run at the same time in the pool.

A maximum of 10 storage pool definitions can be specified for the subsystem description being created. Although each subsystem description can have as many as 10, there is an operational limitation on how many active storage pools can be in the system. In the system, no more than 64 storage pools can be active at any time, including the base storage pool and the machine storage pool. (A storage pool for which *NOSTG has been specified is not considered active, and it is not allocated to any subsystem.)

If a subsystem is started for which all of its storage pools cannot be allocated without exceeding the 64-pool system maximum, the pools that can be allocated (up to the limit) are allocated and the remainder are not. Then, for each routing step started by that subsystem that normally is routed into one of the pools that was not allocated, the base pool is used instead.
This is a required parameter.

You can specify 10 values for this parameter.

**Element 1: Pool identifier**

**1-10** Specify the pool identifier of the storage pool definition to be in this subsystem. The attributes of the pool also must be specified by one of the following values. As many as 10 sets of values can be specified here to define as many as 10 storage pools in the subsystem.

**Element 2: Storage size**

**BASE**

The specified pool definition is defined to be the base system pool, which can be shared with other subsystems. The minimum size and activity level of the base pool are specified in the system values QBASPOOL and QBASACTLVL.

**NOSTG**

No storage and no activity level are assigned to the pool at first. (It is inactive.)

**INTERACT**

The specified pool definition is defined to be the shared pool used for interactive work. The size and activity level of the shared pool are specified using the Change Shared Storage Pool (CHGSHRPOOL) command.

**SPOOL**

The specified pool definition is defined to be the shared pool used for spooled writers. The size and activity level of the shared pool are specified using the CHGSHRPOOL command.

**SHRPOOLnn**

The specified pool definition is defined to be a general-purpose shared pool. There are sixty general-purpose shared pools, identified by special values *SHRPOOL1 to *SHRPOOL60. The size and activity level of a shared pool are specified using the CHGSHRPOOL command.

**integer-number**

Specify the storage size (in kilobytes) of the specified storage pool. A value of at least 256 (meaning 256k) must be specified.

**Element 3: Activity level**

**integer-number**

Specify the maximum number of threads that can run at the same time in the pool.

---

**Maximum jobs (MAXJOBS)**

Specifies the maximum number of jobs that can be active at the same time in the subsystem controlled by this subsystem description. The maximum applies to all jobs that are started and are waiting or running, except for jobs on the job queue or jobs that have finished running.

**NOMAX**

There is no maximum number of jobs in this subsystem.

**0-1000** Specify the maximum number of jobs allowed in this subsystem.
Text ’description’ (TEXT)

Specifies the text that briefly describes the object.

*BLANK

No text is specified.

character-value

Specify no more than 50 characters of text, enclosed in apostrophes.

Sign-on display file (SGNDSPF)

Specifies the name and library of the sign-on display file that is used when showing sign-on displays at work stations allocated to the subsystem. If the specified sign-on display file does not exist when the subsystem description is created or changed, you must specify a library qualifier because the qualified sign-on display file name is kept by the system. The sign-on display file must contain a record format named SIGNON.

Note: The sign-on display file can be changed when the subsystem is active. However, the new sign-on display file is not used until the next time the subsystem is started.

Note: If the user invoking this command has use (*USE) authority to the display file and execute (*EXECUTE) authority to its library, format checks of the display file can be made. This helps predict that the display will work correctly when the subsystem is started. Otherwise, those format checks will not be performed.

Single values

*QDSIGNON

The sign-on display file value QDSIGNON in QSYS is used when showing sign-on displays at work stations that are allocated to the subsystem.

Qualifier 1: Sign-on display file

name

Specify the name of the sign-on display file that is used.

Qualifier 2: Library

*LIBL

All libraries in the thread’s library list are searched until a match is found.

*CURLIB

The current library for the thread is used to locate the object. If no library is specified as the current library for the thread, the QGPL library is used.

name

Specify the library where the sign-on display file is located.

Subsystem library (SYSLIBLE)

Specifies a library that is entered ahead of other libraries in the system portion of the library list. This parameter allows you to use a secondary language library.

Restrictions:
1. The secondary language library should not be specified in the QSYSILIBL or QUSRILIBL system values. QSYSILIBL must contain fewer than 15 libraries to allow the secondary language library to be added to the system portion of the library list.

2. You must have *ALLOBJ and *SECADM special authority to specify a a value other than *NONE for a system library list entry.

*NONE
   The system library list is not changed.

name  Specify the name of the library being added to the system library list.

---

**Authority (AUT)**

Specifies the authority you are giving to users who do not have specific authority for the object, who are not on an authorization list, and whose group profile or supplemental group profiles do not have specific authority for the object.

*LIBCRTAUT
   The authority to the object is the same as the value specified on the Create authority (CRTAUT) parameter of the library in which the object is being created. If the value specified on the CRTAUT parameter is changed, the new value will not affect any existing objects.

*CHANGE
   The user can perform all operations on the object except those limited to the owner or controlled by object existence (*OBJEXIST) and object management (*OBJMGT) authorities. The user can change and perform basic functions on the object. *CHANGE authority provides object operational (*OBJOPR) authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove users.

*ALL
   The user can perform all operations except those limited to the owner or controlled by authorization list management (*AUTLMGT) authority. The user can control the object’s existence, specify the security for the object, change the object, and perform basic functions on the object. The user also can change ownership of the object.

*USE
   The user can perform basic operations on the object, such as displaying its contents. The user cannot change the object. *USE authority provides object operational authority, read authority, and execute authority.

*EXCLUDE
   The user cannot access the object.

name  Specify the name of an authorization list to be used for authority to the object. Users included in the authorization list are granted authority to the object as specified in the list. The authorization list must exist when the object is created.

---

**Examples**

Example 1: Creating a Description With a Signon Display File
This command creates a subsystem description named BAKER and stores it in the current library. If there is no current library, then it is stored in the general purpose library (QGPL). Storage pool definition 1 specifies that pool 1 is to share the base system pool; the definition of storage pool 2 is to have 2000K of storage and an activity level of 4. There is no limit in this subsystem description on the number of jobs that can be active at the same time. The activity levels in the subsystem may, however, be controlled by MAXACT parameters specified in work station entries, job queue entries, and routing entries that are in the subsystem. The sign-on display file is NEWSGNON and is used when showing sign-on displays at work stations allocated to the BAKER subsystem. The user’s library list is searched for the NEWSGNON display file.

**Example 2: Creating a Description that Contains Three Storage Pool Definitions**

This command creates a subsystem description named MEDICAL and stores it in the MEDLIB library. The subsystem description contains three storage pool definitions: storage pool 1 is defined to have 1500K of storage and an activity level of 2, pool 2 is to share the base system pool, and pool 3 is defined first to be inactive when the other pools are active—it has no storage and no activity level. Up to five jobs can be active at the same time in this subsystem. A text description briefly describes the subsystem.

### Error messages

**ESC**APE Messages

CPF1696

Subsystem description &1 not created.
Create Search Index (CRTSCHIDX)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Create Search Index (CRTSCHIDX) command creates a search index. A search index is used to refer to the help information contained in one or more panel groups.

You can access a search index through data description specifications (DDS) by pressing the HELP key, or through the index search function using the Start Search Index (STRSCHIDX) command.

A search index created by the CRTSCHIDX command does not contain any data. Add data using the Add Search Index Entry (ADDSCHIDXE) command.

Restrictions:
- You must have add (*ADD) authority for the library where the search index is to be located.
- IBM-supplied panel groups cannot be added to a search index created with the CRTSCHIDX command; only panel groups created with the Create Panel Group (CRTPNLGRP) command can be added to a search index created with the CRTSCHIDX command.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>SCHIDX</td>
<td>Search index</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Search index</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>TITLE</td>
<td>Display title</td>
<td>Character value</td>
<td>Required, Positional 2</td>
</tr>
<tr>
<td>TEXT</td>
<td>Text ‘description’</td>
<td>Character value, *TITLE, *BLANK</td>
<td>Optional</td>
</tr>
<tr>
<td>CHRID</td>
<td>Character identifier</td>
<td>Single values: *SYSVAL</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Element 1: Graphic character set</td>
<td>Integer</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Code page</td>
<td>Integer</td>
<td></td>
</tr>
</tbody>
</table>

Search index (SCHIDX)

Specifies the search index to be created.

This is a required parameter.

Qualifier 1: Search index

name Specify the name of the search index.
Qualifier 2: Library

*CURLIB

The current library for the job is used to create the search index. If no library is specified as the current library for the job, QGPL is used.

name

Specify the name of the library where the search index is to be created.

Display title (TITLE)

Specifies the title you want to appear at the top of the selected topics display when the search information is presented.

This is a required parameter.

character-value

Specify no more than 55 characters, enclosed in apostrophes.

Text 'description' (TEXT)

Specifies the text that briefly describes the object.

*TITLE

The first 50 characters of the title are used as the text.

*BLANK

No text is specified.

character-value

Specify no more than 50 characters of text, enclosed in apostrophes.

Character identifier (CHRID)

Specifies the graphic character set and code page values used for the search index. The value specified for this parameter must match the TXTCHRID parameter value of panel groups added to this search index.

Single values

*SYSVAL

The system determines the graphic character set and code page values for the command parameters from the QCHRID system value.

Element 1: Graphic character set

integer

Specify the graphic character set value that matches the character set of the synonyms that will be used in the search index.

Element 2: Code page

integer

Specify the code page value that matches the code page of the synonyms that will be used in the search index.
Authority (AUT)

Specifies the authority you are giving to users who do not have specific authority for the object, who are not on an authorization list, and whose group profile or supplemental group profiles do not have specific authority for the object.

*LIBCRTAUT

The system determines the authority for the object by using the value specified for the Create authority (CRTAUT) parameter on the Create Library command (CRTLIB) for the library containing the object to be created. If the value specified for the CRTAUT parameter is changed, the new value will not affect any existing objects.

*CHANGE

The user can perform all operations on the object except those limited to the owner or controlled by object existence (*OBJEXIST) and object management (*OBJMGT) authorities. The user can change and perform basic functions on the object. *CHANGE authority provides object operational (*OBJOPR) authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove users.

*ALL

The user can perform all operations except those limited to the owner or controlled by authorization list management (*AUTLMGT) authority. The user can control the object’s existence, specify the security for the object, change the object, and perform basic functions on the object. The user also can change ownership of the object.

*USE

The user can perform basic operations on the object, such as running a program or reading a file. The user cannot change the object. Use (*USE) authority provides object operational (*OBJOPR), read (*READ), and execute (*EXECUTE) authorities.

*EXCLUDE

The user cannot access the object.

name

Specify the name of an authorization list to be used for authority to the object. Users included in the authorization list are granted authority to the object as specified in the list. The authorization list must exist when the object is created.

Examples

CRTSCHIDX SCHIDX(ACCOUNTING) TITLE('Accounting Help Index')
   TEXT('Accounting Help Index')

This command creates a search index named ACCOUNTING in the current library.

Error messages

*ESCAPE Messages

CPF6E11

Search index &2 not created in library &3.
Create Spelling Aid Dictionary (CRTSPADCT)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Create Spelling Aid Dictionary (CRTSPADCT) command allows the user to create a user-defined spelling aid dictionary. This dictionary is used with the Create Document (CRTDOC), Check Document (CHKDOC), Edit Document (EDTDOC), or Work with Documents (WRKDOC) commands when doing spell check functions, such as spelling aid and spelling verification.

Spelling aid dictionaries are classified into IBM language dictionaries and user dictionaries. Language dictionaries are IBM-created; user dictionaries are created with this command.

The following are IBM language dictionary names that are in library QDCT. They should not be used for user-created dictionaries.

- AFRIKAAN
- AKTUEEL
- BRASIL
- CATALA
- DANSK
- DEUTSCH
- DSCHWEIZ
- ESPANA
- FRANCAIS
- FRA2
- GREEK
- ISLENSK
- ITALIANO
- LEGAL
- MEDICAL
- NEDERLND
- NORBOK
- NORNYN
- PORTUGAL
- RUSSIAN
- SUOMI
- SVENSK
- UK
- US

This command requires a source file to contain all the words being put into the spelling aid dictionary. The source member can be created by using the Start Source Entry Utility (STRSEU) command. The dictionary source type is SPADCT.
## Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
</table>
| SPADCT   | Spelling aid dictionary      | Single values: *USRSWL  
Other values: Qualified object name                                      | Required, Positional 1       |
|          | Qualified 1: Spelling aid    | Name                                                                    |                              |
|          | dictionary                  |                                                                         |                              |
|          | Qualified 2: Library         | Name, *CURLIB                                                          |                              |
| SRCFILE  | Source file                  | Qualified object name                                                  | Required, Positional 2       |
|          | Qualified 1: Source file      | Name                                                                    |                              |
|          | Qualified 2: Library         | Name, *LIBL, *CURLIB                                                   |                              |
| SRCMBR   | Source member                | Name, *DCT                                                             | Optional, Positional 3       |
| TEXT     | Text ‘description’           | Character value, *SRCMBRTEXT, *BLANK                                    | Optional                     |
| BASEDCT  | Base dictionary              | Qualified object name                                                  | Optional                     |
|          | Qualified 1: Base dictionary | Name, *NONE                                                            |                              |
|          | Qualified 2: Library         | Name, *LIBL, *CURLIB                                                   |                              |
| VFYDCT   | Verify dictionary            | Qualified object name                                                  | Optional                     |
|          | Qualified 1: Verify dictionary| Name, *NONE                                                            |                              |
|          | Qualified 2: Library         | Name, *LIBL, *CURLIB                                                   |                              |
| SWLLANGID| Stop word list language ID   | Character value                                                        | Optional                     |
| BASESWL  | Base stop word list          | *IBM, *NONE                                                            | Optional                     |
| OPTION   | Source listing option        | *SOURCE, *NOSOURCE, *SRC, *NOSRC                                       | Optional, Positional 4       |
| REPLACE  | Replace dictionary           | *YES, *NO                                                              | Optional                     |

### Spelling aid dictionary (SPADCT)

Specifies the name and library for the spelling aid dictionary being created.

This is a required parameter.

**USRSWL**

A user-defined stop word list is created using an IBM-supplied name.

The possible library values are:

**CURLIB**

The current library for the job is used to store the dictionary. If no library is specified as the current library for the job, QGPL is used.

**library-name**

Specify the name of the library where the dictionary is stored.
Source file (SRCFILE)

Specifies the name and library of the source file used when the spelling aid dictionary is created. The source file contains the source member that is used for creating the dictionary.

This is a required parameter.

The possible library values are:

*LIBL    All libraries in the library list for the current thread are searched until the first match is found.

*CURLIB   The current library for the job is used to locate the source file. If no library is specified as the current library for the job, QGPL is used.

library-name    Specify the name of the library in which the source file is located.

Source member (SRCMBR)

Specifies the name of the source file member that contains the words for the dictionary being created. The member is located in the source file specified on the Source file prompt (SRCFILE parameter).

*DCT    The source file member name is the same as that of the dictionary being created.

source-file-member-name    Specify the name of the member in the source file that is used to create the spelling aid dictionary. A member name must be specified when the source file member being processed does not have the same name as the spelling aid dictionary being created.

Text ’description’ (TEXT)

Specifies the text that briefly describes the object.

*SRCMBRTEXT    The text is taken from the source file member being used to create the spelling aid dictionary.

’description’    Specify no more than 50 characters of text, enclosed in apostrophes.

*BLANK    No text is specified.

Base dictionary (BASEDCT)

Specifies the name and library of the dictionary that contains words that are added to the dictionary being created. An IBM language dictionary (one created by IBM) cannot be used here.

*NONE    No words from another dictionary are added to the dictionary being created.
Specify the name and library of the dictionary that contains words to be added to the dictionary being created.

The possible library values are:

*LIBL All libraries in the library list for the current thread are searched until the first match is found.
*CURLIB The current library for the job is used to locate the dictionary. If no library is specified as the current library for the job, QGPL is used.

Specify the name of the library where the dictionary is located.

Verify dictionary (VFYDCT)

Specifies the name and library of an existing dictionary that is searched for each word specified in the source member. Only those words that are not found in the existing dictionary are placed in the new dictionary to avoid duplication.

*NONE Every word specified in the source member is placed in the new spelling aid dictionary without verifying against another dictionary.

dictionary-name Specify the name of the dictionary that contains words that are not to be duplicated in the dictionary being created.

The possible library values are:

*LIBL All libraries in the library list for the current thread are searched until the first match is found.
*CURLIB The current library for the job is used to locate the dictionary. If no library is specified as the current library for the job, QGPL is used.

library-name Specify the name of the library where the dictionary is located.

Language attribute (LNGATR)

Specifies the language attribute associated with the dictionary being created. The language attribute determines the processing rules that apply when the dictionary is used.

*VFYDCT The language attribute of the dictionary is the same as the language attribute of the dictionary that is specified on the Verify dictionary prompt (VFYDCT parameter).

*NONE The dictionary being created does not have a specific language attribute.

*AFRIKAAN The language attribute used by the spelling aid dictionary is Afrikaans.

*BRASIL The language attribute used by the spelling aid dictionary is Brazilian Portuguese.
The language attribute used by the spelling aid dictionary is Catalan.

The language attribute used by the spelling aid dictionary is Danish.

The language attribute used by the spelling aid dictionary is German.

The language attribute used by the spelling aid dictionary is Swiss-German.

The language attribute used by the spelling aid dictionary is English.

The language attribute used by the spelling aid dictionary is Spanish.

The language attribute used by the spelling aid dictionary is French.

The language attribute used by the spelling aid dictionary is French, where accents are required on uppercase characters.

The language attribute used by the spelling aid dictionary is Greek.

The language attribute used by the spelling aid dictionary is Icelandic.

The language attribute used by the spelling aid dictionary is Italian.

The language attribute used by the spelling aid dictionary is Dutch.

The language attribute used by the spelling aid dictionary is Norwegian.

The language attribute used by the spelling aid dictionary is Portuguese.

The language attribute used by the spelling aid dictionary is Russian.

The language attribute used by the spelling aid dictionary is Finnish.

The language attribute used by the spelling aid dictionary is Swedish.

The language attribute used by the spelling aid dictionary is Turkish.

Stop word list language ID (SWLLANGID)

Specifies the language identifier (ID) for the stop word list.
Base stop word list (BASESWL)

Specifies whether the IBM-supplied stop word list words are included in the user-created stop word list.

*IBM  The words from the IBM-supplied stop word list are added to the user-created stop word list.

*NONE  No words from the IBM-supplied stop word list are added to the user-created stop word list.

*Note: Word entries in the IBM-supplied stop word list source file preceded by a dash, (&ndash.), are not added to the user-created stop word list even when BASESWL(*IBM) is specified.

Source listing option (OPTION)

Specifies the type of output listing that is produced when the dictionary is created.

*SRC or *SOURCE  A listing of the source statements that are used to create the dictionary, as well as a listing of any errors that occur, is created.

*NOSRC or *NOSOURCE  No listing of the source statements is generated unless errors occur.

Authority (AUT)

Specifies the authority you are giving to users who do not have specific authority for the object, who are not on an authorization list, and whose group profile or supplemental group profiles do not have specific authority for the object.

*LIBCRTAUT  The authority for the object is the same as the value specified on the Create authority prompt (CRTAUT parameter) of the library in which the object is being created. If the value specified on the Create authority prompt (CRTAUT parameter) is changed, the new value will not affect any existing objects.

*CHANGE  The user can change the dictionary and use it to check the spelling of the content of a text document.

*ALL  The user can perform all operations except those limited to the owner or controlled by authorization list management (*AUTLMGT) authority. The user can control the object’s existence, specify the security for the object, change the object, and perform basic functions on the object. The user also can change ownership of the object.

*USE  The user can perform basic operations on the object, such as running a program or reading a file. The user cannot change the object. Use (*USE) authority provides object operational (*OBJOPR), read (*READ), and execute (*EXECUTE) authorities.

*EXCLUDE  The user cannot access the object.
**authorization-list-name**

Specify the name of an authorization list to be used for authority to the object. Users included in the authorization list are granted authority to the object as specified in the list. The authorization list must exist when the object is created.

### Replace dictionary (REPLACE)

Specifies whether an existing dictionary is replaced when a new dictionary is created using the same name as an existing dictionary.

**YES**  The dictionary being created replaces an existing dictionary with the same name.

**NO**   The dictionary being created does not replace an existing dictionary with the same name.

### Examples

**CRTSPADCT**

<table>
<thead>
<tr>
<th>Command</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>SPADCT(MYLIB/MYDCT)</td>
<td>SRCFILE(MYLIB/SRC)</td>
</tr>
<tr>
<td>SRCMBR(WORDS)</td>
<td>OPTION(*NOSRC)</td>
</tr>
<tr>
<td>BASEDCT(QGPL/BASDCT)</td>
<td>VFYDCT(QDCT/US)</td>
</tr>
<tr>
<td>LNGATR(*ENGLISH)</td>
<td>REPLACE(*YES)</td>
</tr>
</tbody>
</table>

This command creates a spelling aid dictionary named MYDCT in the library MYLIB. The words used in the spelling aid dictionary are from source member WORDS of the SRC source file in MYLIB. The dictionary includes words from a dictionary named BASDCT, but does not contain any words that are found in the dictionary named US. The dictionary being created has the *ENGLISH attribute. If an existing dictionary is named MYDCT, it is replaced.

### Error messages

***ESCAPE Messages**

**CPF2283**

Authorization list &1 does not exist.

**CPF4102**

File &2 in library &3 with member &4 not found.

**CPF4104**

User not authorized to operation on file &2 in &3, member, device, or program device &4.

**CPF411B**

Shared open of member &4 not successful.

**CPF9801**

Object &2 in library &3 not found.

**CPF9802**

Not authorized to object &2 in &3.

**CPF9803**

Cannot allocate object &2 in library &3.

**CPF9810**

Library &1 not found.
CPF9820
Not authorized to use library &1.

CPF9832
Function not supported for DDM file &2.

CPF9845
Error occurred while opening file &1.

CPF9848
Cannot open file &1 in library &2 member &3.

CPF9899
Error occurred during processing of command.
Create SQL Package (CRTSQLPKG)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Create Structured Query Language Package (CRTSQLPKG) command allows you to create (or re-create) an SQL package on a relational database from an existing distributed SQL program. A distributed SQL program is a program created by specifying the Relational database (RDB) parameter on a CRTSQLxxx (where xxx = CBL, CBLI, CI, CPPI, PLI, RPG or RPGI) command.

More information is in the SQL Programming information in the iSeries Information Center at http://www.ibm.com/eserver/iseries/infocenter.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>PGM</td>
<td>Program</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td>QUAL1</td>
<td>Program</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td>QUAL2</td>
<td>Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>RDB</td>
<td>Relational database</td>
<td>Simple name, *PGM</td>
<td>Optional, Positional 2</td>
</tr>
<tr>
<td>USER</td>
<td>RDB user</td>
<td>Name, *CURRENT</td>
<td>Optional</td>
</tr>
<tr>
<td>PASSWORD</td>
<td>RDB user password</td>
<td>Character value, *NONE, ' '</td>
<td>Optional</td>
</tr>
<tr>
<td>DFTRDBCOL</td>
<td>Default collection</td>
<td>Name, *PGM, *NONE</td>
<td>Optional</td>
</tr>
<tr>
<td>OBJTYPE</td>
<td>Object type</td>
<td>*PGM, *SRVPGM</td>
<td>Optional</td>
</tr>
<tr>
<td>MODULE</td>
<td>Module list</td>
<td>Single values: *ALL Other values (up to 256 repetitions): Name</td>
<td></td>
</tr>
<tr>
<td>TEXT</td>
<td>Text ‘description’</td>
<td>Character value, *PGMTXT, *BLANK</td>
<td>Optional</td>
</tr>
<tr>
<td>GENLVL</td>
<td>Severity level</td>
<td>0-40, 10</td>
<td>Optional</td>
</tr>
<tr>
<td>REPLACE</td>
<td>Replace</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
<tr>
<td>PRTFILE</td>
<td>Print file</td>
<td>Qualified object name</td>
<td>Optional</td>
</tr>
<tr>
<td>QUAL1</td>
<td>Print file</td>
<td>Name, QSYSPR</td>
<td></td>
</tr>
<tr>
<td>QUAL2</td>
<td>Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
</tr>
</tbody>
</table>

Program (PGM)

Specifies the name of the program for which the SQL package is being created. The program must be a distributed SQL program that was created using one of the CRTSQLxxx (where xxx = CBL, CBLI, CI, CPPI, PLI, RPG or RPGI) commands.

Qualifier 1: Program

name Specify the name of the program for which the SQL package is to be created.
Qualifier 2: Library

*LIBL All libraries in the library list for the current thread are searched until the first match is found.

*CURLIB The current library for the job is used to locate the program. If no current library is specified as the current library for the job, QGPL is used.

name Specify the name of the library where the program is located.

Relational database (RDB)

Specifies the relational database where the SQL package is being created.

*PGM The relational database name specified for the Relational database (RDB) parameter of the CRTSQLxxx (where xxx = CBL, CBIL, CI, CPPI, PLI, RPG or RPGI) command that originally created the program is used.

name Specify the name of the relational database where the SQL package is to be created. Use the Work with Relational Database Directory Entry (WRKRDBDIRE) command to show the relational database names that are valid for this parameter.

RDB user (USER)

Specifies the user name sent to the remote system when starting the conversation.

*CURRENT The user name associated with the current job is used.

name Specify the user name being used for the application requester job.

RDB user password (PASSWORD)

Specifies the password to be used on the remote system.

*NONE No password is sent. The user name specified for the RDB user (USER) parameter is not valid if this value is specified.

character-value Specify the password of the user name specified for the USER parameter. A password value of a blank is treated the same as specifying *NONE.

Default collection (DFTRDBCOL)

Specifies the schema name to be used for unqualified names of tables, views, indexes, SQL packages, aliases, constraints, external programs, node groups, and triggers. This parameter applies only to static SQL statements in the package.

*PGM The schema name specified for the Default collection (DFTRDBCOL) parameter of the
CRTSQLxxx (where xxx = CBL, CBLI, CI, CPPI, PLI, RPG or RPGI) command used to create the program for which an SQL package is being created is used.

*NONE
The Naming convention option specified for the Precompiler options (OPTION) parameter of the CRTSQLxxx (where xxx = CBL, CBLI, CI, CPPI, PLI, RPG or RPGI) command used to precompile the program is used to determine the schema name.

name Specify the schema name that is used for unqualified tables, views, indexes, SQL packages, aliases, constraints, external programs, node groups, and triggers.

Object type (OBJTYPE)
Specifies the type of program for which an SQL package is created.

*PGM Create an SQL package from the program specified for the Program (PGM) parameter.
*SRVPGM Create an SQL package from the service program specified for the PGM parameter.

Module list (MODULE)
Specifies a list of modules in a bound program.

Single values
*ALL An SQL package is created for all modules in the program. An error message is sent if none of the modules in the program contain SQL statements or none of the modules is a distributed program.

Other values (up to 256 repetitions)

name Specify the modules in the program for which an SQL package is to be created. If more than 256 modules exist that need to be packaged, multiple CRTSQLPKG commands must be used. A maximum of 1024 modules can be in a program that has at least one module containing an SQL statement.

Duplicate module names in the same program are allowed. This command looks at each module in the program and if *ALL or the module name is specified for the MODULE parameter, processing continues to determine whether an SQL package should be created. If the module is created using SQL and the Relational database (RDB) parameter is specified on the precompile command, an SQL package is created for the module. The SQL package is associated with the module of the bound program.

Text 'description' (TEXT)
Specifies text that briefly describes the SQL package and its function.

*PGMTXT The text from the program for which the SQL package is being created is used.
*BLANK No text is specified.
'description'
   Specify no more than 50 characters of text, enclosed in apostrophes.

Severity level (GENLVL)
Specifies the maximum severity level allowed for errors detected during SQL package creation. If errors occur at a level that exceeds the level you specify, the SQL package is not created.

10  The maximum severity level is 10.
0-40 Specify the maximum severity level.

Replace (REPLACE)
Specifies whether an existing SQL package of the same name in the specified library is replaced by the new SQL package.

*YES  An existing SQL package of the same name is replaced by the new SQL package.
*NO   An existing SQL package of the same name is not replaced; a new SQL package is not created if the package already exists.

Print file (PRTFILE)
Specifies the printer device file to which the create SQL package error listing is directed. If no errors are detected during the creation of the SQL package, no listing is produced.

Qualifier 1: Print file

QSYSPR
   The create SQL package error listing is directed to the IBM-supplied printer file, QSYSPR.
name   Specify the name of the printer device file to which the create SQL package error listing is directed.

Qualifier 2: Library

*LIBL  All libraries in the library list for the current thread are searched until the first match is found.
*CURLIB The current library for the job is used to locate the printer file. If no library is specified as the current library for the job, QGPL is used.
name   Specify the name of the library where the printer file is located.

Examples

CRTSQLPKG   PGM(PAYROLL)  RDB(SYSTEMA)  TEXT('Payroll Program')

This command creates an SQL package from the distributed SQL program PAYROLL on relational database SYSTEMA.
**Error messages**

*ESCAPE Messages*

**SQL9004**
Create of SQL package failed.

**SQL9006**
DB2 UDB Query Manager and SQL Development Kit for iSeries not at same install level as i5/OS.
IBM Systems - iSeries: i5/OS Commands Starting with CRTMSGF (Create Message File)
Create Source Physical File (CRTSRCPF)

Where allowed to run: All environments (*ALL)
Threadsafe: Conditional

The Create Source Physical File (CRTSRCPF) command creates a source physical file.

A source physical file contains source data needed to create objects such as control language (CL) source statements, which are used to create a CL program, or data description specifications (DDS) which (in turn) are used to create a database or device file.

A source physical file can have one or more members. The maximum number of members that can be added to the file is specified for the **Maximum members (MAXMBRS)** parameter.

**Restrictions:**
- This command is conditionally threadsafe. In multithreaded jobs, this command is not threadsafe and fails for Distributed Data Management (DDM) files of type *SNA, when SYSTEM(*RMT) or SYSTEM(*FILETYPE) is specified.

**Parameters**

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FILE</strong></td>
<td>File</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: File</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *CURLIB</td>
<td></td>
</tr>
<tr>
<td><strong>RCDLEN</strong></td>
<td>Record length</td>
<td>Integer, 92</td>
<td>Optional, Positional 2</td>
</tr>
<tr>
<td><strong>MBR</strong></td>
<td>Member, if desired</td>
<td>Name, *NONE, *FILE</td>
<td>Optional, Positional 3</td>
</tr>
<tr>
<td><strong>IGCDTA</strong></td>
<td>User specified DBCS data</td>
<td>*NO, YES</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>TEXT</strong></td>
<td>Text 'description'</td>
<td>Character value, *BLANK</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>SYSTEM</strong></td>
<td>System</td>
<td>*LCL, *RMT, *FILETYPE</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>EXPDATE</strong></td>
<td>Expiration date for member</td>
<td>Date, *NONE</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>MAXMBRS</strong></td>
<td>Maximum members</td>
<td>Integer, *NOMAX</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>ACCPTH SZ</strong></td>
<td>Access path size</td>
<td>*MAXITB, *MAX4GB</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>PAGESIZE</strong></td>
<td>Access path logical page size</td>
<td>*KEYLEN, 8, 16, 32, 64, 128, 256, 512</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>ACCPTH</strong></td>
<td>Access path type</td>
<td>*ARRIVAL, *KEYED</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>MAINT</strong></td>
<td>Access path maintenance</td>
<td>*IMMED, *DLY, *REBLD</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>RECOVER</strong></td>
<td>Access path recovery</td>
<td>*NO, *AFTIPL, *IPL</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>FRCA</strong></td>
<td>Force keyed access path</td>
<td>*NO, *YES</td>
<td>Optional</td>
</tr>
</tbody>
</table>
Keyword | Description | Choices | Notes |
---|---|---|---|
**SIZE** | Member size | Single values: "NOMAX" Other values: "Element list" | Optional |
| | Element 1: Initial number of records | 1-2147483646, **10000** | |
| | Element 2: Increment number of records | **Integer, 1000** | |
| | Element 3: Maximum increments | **Integer, 499** | |
**ALLOCATE** | Allocate storage | "NO, "YES | Optional |
**CONTIG** | Contiguous storage | "NO, "YES | Optional |
**UNIT** | Preferred storage unit | 1-255, "ANY | Optional |
**FRCRATIO** | Records to force a write | **Integer, "NONE** | Optional |
**WAITFILE** | Maximum file wait time | **Integer, "IMMED, "CLS** | Optional |
**WAITRCD** | Maximum record wait time | **Integer, 60, "IMMED, "NONE** | Optional |
**SHARE** | Share open data path | "NO, "YES | Optional |
**DLTPCT** | Max % deleted records allowed | 1-100, "NONE** | Optional |
**CCSID** | Coded character set ID | **Integer, "JOB, "HEX** | Optional |
**ALWUPD** | Allow update operation | "YES, "NO | Optional |
**ALWDLT** | Allow delete operation | "YES, "NO | Optional |
**AUT** | Authority | Name, "LIBCRTAUT, "ALL, "CHANGE, "EXCLUDE, "USE | Optional |

**File (FILE)**

Specifies the source physical file to be created.

If the file is used in a high-level language program, the file name must be consistent with the naming rules of that language. Otherwise, the file must be renamed in the program.

This is a required parameter.

**Qualifier 1: File**

name Specify the name of the source physical file.

**Qualifier 2: Library**

*CURLIB*

The current library for the job is used to locate the file. If no library is specified as the current library for the job, QGPL is used.

name Specify the library where the file is located.

**Record length (RCDLEN)**

Specifies the number of bytes in the length of the records stored in the source physical file. The record format contains three fields: the source sequence number, the date, and the source statement.
This parameter must provide 12 positions for the source sequence number and date fields required in each record. These fields are defined with fixed attributes and names, and they have a keyed access path over the sequence number.

integer

Specify the record length of each source record in the file. The value must include 6 bytes for the source sequence number and 6 bytes for the date. Valid values range from 13 through 32766 bytes.

Double-Byte Character Set Considerations

If IGCDTA(*YES) is specified, the RCDLEN parameter must provide six positions for the source sequence number, six positions for the date field, and at least four positions for source start. Valid values for a double-byte character set (DBCS) range from 16 through 32766.

Member (MBR)

Specifies the source file member to be added when the file is created.

*NONE

No member is added when the file is created.

*FILE

The name of the member to be added is the same as the name specified for the File (FILE) parameter.

name

Specify the name of the member to be added.

User specified DBCS data (IGCDTA)

Specifies whether the file contains double-byte character set (DBCS) data.

*NO

The file does not contain DBCS data.

*YES

The file contains DBCS data.

Text ’description’ (TEXT)

Specifies the text that briefly describes the object.

*BLANK

No text is specified.

character-value

Specify no more than 50 characters of text, enclosed in apostrophes.

System (SYSTEM)

Specifies whether the source physical file is created on the local system or the remote system.
*LCL  The source physical file is created on the local system. The file specified for the File (FILE) parameter must not already exist on the system.

*RMT  The source physical file is created on a remote system. The file specified for the FILE parameter must be the name of a DDM file that identifies the remote system and the name of the source physical file being created.

*FILETYPE
If the file specified for the FILE parameter does not exist on the system, the source physical file is created on the local system. Otherwise, the file on the FILE parameter must be a DDM file, and the source physical file is created on a remote system. The DDM file identifies the remote system and the name of the source physical file being created.

Expiration date for member (EXPDATA)
Specifies, if a source file member is added when the source physical file is created, the expiration date of the source file member.

*NONE  The member being added to the file has no expiration date.

date  Specify the date after which the member being added cannot be used.

Maximum members (MAXMBRS)
Specifies the maximum number of members that the source physical file can contain.

*NOMAX  The number of members that can be contained in the file is the system maximum of 32,767 members.

integer  Specify the maximum number of members that can be contained in the file. Valid values range from 1 through 32767 members.

Access path size (ACCPTHGSIZ)
Specifies the maximum size of auxiliary storage that can be occupied by access paths that are associated with keyed source physical files. This parameter does not apply to access paths that are created for logical files or for queries that refer to the data in a source physical file.

*MAX1TB  The access paths associated with this file can occupy a maximum of one terabyte (1,099,511,627,776 bytes) of auxiliary storage.

*MAX4GB  The access paths associated with this file can occupy a maximum of four gigabytes (4,294,966,272 bytes) of auxiliary storage.
**Access path logical page size (PAGESIZE)**

Specifies the access path logical page size that is used when the access path is created.

The access path logical page size is used by the system to determine the size of each page of the index. This logical page size is the amount of bytes of the access path that can be moved into the job’s storage pool from the auxiliary storage for a page fault.

*KEYLEN

The access path logical page size will be determined by the total length of the key, or keys.

- 8 Logical page size of 8k.
- 16 Logical page size of 16k.
- 32 Logical page size of 32k.
- 64 Logical page size of 64k.
- 128 Logical page size of 128k.
- 256 Logical page size of 256k.
- 512 Logical page size of 512k.

**Access path type (ACCPTH)**

Specifies the type of access path used by all the members in the source physical file.

*ARRIVAL

The access path is an arrival sequence access path.

*KEYED

The access path is a keyed sequence access path.

**Access path maintenance (MAINT)**

Specifies the type of access path maintenance used for all members of the source physical file.

*IMMED

The access path is updated each time a record is changed, added, or deleted from a member.

*REBLD

The access path is completely rebuilt each time a file member is opened. The access path is maintained until the member is closed. Then the access path is deleted.

*DLY

The maintenance of the access path is delayed until the physical file member is opened for use. The access path is changed only for records that have been added, deleted, or changed since the file was last opened. While the file is open, all changes made to its members are immediately reflected in the access path of those members, no matter what is specified for the MAINT parameter. To prevent a lengthy rebuild time when the file is opened, *DLY should be specified only when the number of changes to the access path is small.

If the number of changes between a close and the next open reaches approximately 10 percent of the access path size, the system stops saving changes and the access path is completely rebuilt the next time the file is opened.
Access path recovery (RECOVER)

Specifies, for files having immediate or delayed maintenance on their access paths, when recovery processing of the file is performed after a system failure occurs while the access path is being changed. This parameter is valid only for files with a keyed access path.

If *IMMED or *DLY is specified for the Access path maintenance (MAINT) parameter, the access path can be rebuilt during initial program load (IPL) (before any user can run a job), after IPL has ended (during concurrent job running), or when the file is next opened. While the access path is being rebuilt, the file cannot be used by any job.

During the IPL, an Override Access Path Recovery display lists those paths that must be recovered and what the RECOVER parameter value is for each path. The user can override the RECOVER parameter value on this display. More information is in the Backup and Recovery book, SC41-5304.

If *REBLD is specified for the MAINT parameter, the access path is rebuilt the next time its file is opened.

*NO The access path of the file is rebuilt when the file is opened. *NO is the default for all files that do not require unique keys.

*AFTIPL
The access path of the file is rebuilt after the initial program load (IPL) operation is completed. This option allows other jobs not using this file to start processing immediately after the completion of IPL. If a job tries to allocate the file while its access path is being rebuilt, a file open exception occurs. *AFTIPL is the default for files that require unique keys.

*IPL The access path of the file is rebuilt during the IPL operation. This ensures that the file’s access path is rebuilt before the first user program tries to use it; however, no jobs can start running until after all files that specify RECOVER(*IPL) have their access paths rebuilt.

Force keyed access path (FRCACCPTH)

Specifies whether access path changes are forced to auxiliary storage along with the associated records in the source physical file.

*NO The access path and the associated records are not written to auxiliary storage whenever the access path is changed.

*YES The access path and the associated records are written to auxiliary storage whenever the access path is changed. *YES cannot be specified if *REBLD is specified for the Access path maintenance (MAINT) parameter.

Member size (SIZE)

Specifies the initial number of records in each member of the file, the number of records in each part added to the member size, and the number of times the part added is automatically applied. The number of records for each file member is specified as the number of records that can be placed in it (this number includes any deleted records).

When the maximum number of records has been reached, a message (stating that the member is full) is sent to the system operator, giving the choice of ending the request or extending the member size. The
operator can extend the member by 10% or by the number of records specified as the increment value, whichever is greater, each time the message is received.

Single values

*NOMAX
The number of records that can be added to each member of the file is not limited by the user. The maximum size of each member is determined by the system. If *NOMAX is specified, *NO must be specified for the Allocate storage (ALLOCATE) parameter.

Element 1: Initial number of records

Specify the initial number of records in each member.

10000 Initially, up to 10000 records can be written to each member of the file.

1-2147483646 Specify the number of records that can be written to each member of the file before the member size is automatically extended.

Element 2: Increment number of records

Specify the number of records that are automatically added to the member when the number of records in the member is greater than the initial member size. The minimum size of an increment is 10% of the size of the member at the time the maximum number of records is reached.

1000 The file size is increased by 10% or 1000 records, whichever is greater.

integer Specify the number of additional records which, if greater than 10% of the size of the member when the maximum number of records is reached, are automatically added to the member.

If the number specified is not greater than 10% of the member size and not equal to zero, the member size is increased by 10%.

If 0 is the specified increment value, the member is not automatically extended. This value must be 0 if the value for the number of increments is 0.

Element 3: Maximum increments

Specify the maximum number of increments that can be automatically added to the member.

499 A maximum of 499 increments is automatically added to the member size.

integer Specify the maximum number of increments automatically added to the member size. Valid values range from 0 through 32767. If 0 is specified, the member is not automatically extended.

Allocate storage (ALLOCATE)

Specifies whether initial storage space is allocated to each physical file member added to the file. The allocation provides enough space to hold the number of records specified for the Member size (SIZE) parameter. Allocations that occur when a record cannot be added to a member without exceeding its capacity are determined by the system and by the SIZE parameter values.

*NO The system determines the amount of storage space to allocate to each member added to the file.
*YES  The amount of storage space specified in the first value of the SIZE parameter is allocated each time a new member is added. If *YES is specified, *NOMAX must not be specified for the SIZE parameter.

Preferred storage unit (UNIT)

This parameter is no longer supported. It exists solely for compatibility with releases earlier than Version 3 Release 6 Modification 0 of the OS/400 operating system. For information on using auxiliary storage pools (ASPs), refer to the Backup and Recovery book, SC41-5304.

You can specify the value *ANY or a value ranging from 1 through 255 on this parameter.

Records to force a write (FRCRATIO)

Specifies the number of inserted or updated records that are processed before the records are forced into auxiliary storage.

The force write ratio specified for a logical file cannot be less than or equal to the smallest force write ratio of its based-on files. If a larger force write ratio is specified, it is ignored and a message is sent informing the user of the action.

For example, if the force ratios of three physical files are 2, 6, and 8, the logical file force ratio that is based on these three physical files must be as restrictive as the least of them; that is 2 in this case. Two would be used even if the FRCRATIO parameter is not specified. Thus, each time a program inserts, updates, or deletes two records in the logical file (regardless of which based-on physical files are affected), those records are forced to permanent storage.

If a physical file associated with this logical file is being journaled, a large force write ratio or *NONE is specified. More information on journal management is in the Backup and Recovery book, SC41-5304.

*NONE  There is no specified force ratio. The system determines when the records are written to auxiliary storage.

integer  Specify the number of inserted or updated records that are processed before the records are written to auxiliary storage.

Maximum file wait time (WAITFILE)

Specifies the number of seconds that the program waits for the file resources to be allocated when the file is opened, or the device or session resources to be allocated when an acquire operation is performed to the file. If the file resources cannot be allocated in the specified wait time, an error message is sent to the program.

*IMMED  The program does not wait. Immediate allocation of file resources is required.
*CLS The job default wait time is used as the wait time for the file resources to be allocated.

1-32767 Specify the number of seconds to wait for file resources to be allocated.

**Maximum record wait time (WAITRCD)**

Specifies the number of seconds that the program waits for a record being changed or deleted. If the record cannot be allocated within the specified wait time, an error message is sent to the program.

60 The program waits for 60 seconds for a record being changed or deleted.

*IMMED The program does not wait. Immediate allocation of file resources is required.

*NOMAX The wait time is the maximum allowed by the system, which is 32767 seconds.

**integer** Specify the number of seconds that the program waits for a record being changed or deleted. Valid values range from 1 through 32767 seconds.

**Share open data path (SHARE)**

Specifies whether the open data path (ODP) is shared with other programs in the same routing step. When an ODP is shared, the programs accessing the file share facilities such as the file status and the buffer.

**Note:** This parameter cannot be specified when *NONE is specified for the Member (MBR) parameter.

*NO The ODP is not shared with other programs in the routing step. A new ODP for the file is created and used every time a program opens the file.

*YES The same ODP is shared with each program in the job that also specifies *YES when it opens the file.

**Max % deleted records allowed (DLTPCT)**

Specifies the maximum percentage of deleted records for each member in the source physical file. The percentage check is made when the member is closed. If the percentage of deleted records is greater than the value specified on this parameter, a message is sent to the job log.

*NONE The percentage of deleted records in the file members is not checked.

1-100 Specify the largest allowed percentage of deleted records for any member in the file.
Coded character set ID (CCSID)

Specifies the coded character set identifier (CCSID) used to describe character data in the fields of the source file.

*JOB  The current job’s default CCSID is used.

*HEX  The CCSID 65535 is used, which indicates that character data in the fields is treated as bit data and is not converted.

integer  Specify the CCSID to be used.

Allow update operation (ALWUPD)

Specifies whether records in this source physical file can be updated.

*YES  Records in this source file can be updated.

*NO  Records in this source file cannot be updated.

Allow delete operation (ALWDLT)

Specifies whether records in this source physical file can be deleted.

*YES  Records in this source file can be deleted.

*NO  Records in this source file cannot be deleted.

Authority (AUT)

Specifies the authority you are giving to users who do not have specific authority for the object, who are not on an authorization list, and whose group profile or supplemental group profiles do not have specific authority for the object.

*LIBCRTAUT  The system determines the authority for the object by using the value specified for the Create authority (CRTAUT) parameter on the Create Library command (CRTLIB) for the library containing the object to be created. If the value specified for the CRTAUT parameter is changed, the new value will not affect any existing objects.

*CHANGE  The user can perform all operations on the object except those limited to the owner or controlled by object existence (*OBJEXIST) and object management (*OBJMGT) authorities. The user can change and perform basic functions on the object. *CHANGE authority provides object operational (*OBJOPR) authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove users.

*ALL  The user can perform all operations except those limited to the owner or controlled by authorization list management (*AUTLMGT) authority. The user can control the object’s existence, specify the security for the object, change the object, and perform basic functions on the object. The user also can change ownership of the object.

*USE  The user can perform basic operations on the object, such as running a program or reading a file.
The user cannot change the object. Use (*USE) authority provides object operational (*OBJOPR), read (*READ), and execute (*EXECUTE) authorities.

**EXCLUDE**

The user cannot access the object.

*name* Specify the name of an authorization list to be used for authority to the object. Users included in the authorization list are granted authority to the object as specified in the list. The authorization list must exist when the object is created.

---

**Examples**

**Example 1: Creating a File Without Members**

```shell
CRTSRCPF FILE(SRCLIB/PAYTXS)
```

This command creates a source file named PAYTXS in the SRCLIB library. The file is created without any members; therefore, no data can be put into the file until a member is added later. As many as 32,767 members (*NOMAX) can be added to the file.

Each member can have up to 10000 records before automatic extensions (499 increments maximum) occur that add 1000 records to the capacity of the member. Only minimum initial storage is allocated for each member with no restrictions on whether the space is connected. The public has object operational, read, add, delete, and update authority for the file, but no object management or object existence authority.

**Example 2: Creating a File With a Member**

```shell
CRTSRCPF FILE(ORDERCTL/ORDERS) MBR(*FILE) SIZE(100 50 5)
```

This command creates a source physical file named ORDERS in the ORDERCTL library. Storage space for the records placed in the file need not be contiguous. The initial allocation of storage provides for up to 100 records, and up to five increments of additional space for 50 records each can be added automatically. These allocation values also apply to members of this source file that will be added later.

**Example 3: Creating a File that Contains DBCS Data**

```shell
CRTSRCPF FILE(IGCLIB/IGCSRC) IGCDTA(*YES)
```

This command creates a source physical file named IGCSRC, which is stored in the library IGCLIB, and can contain DBCS data.

---

**Error messages**

***ESCAPE Messages**

**CPF323C**

QRECOVERY library could not be allocated.

**CPF5702**

File either not DDM file or not found.

**CPF7302**

File &1 not created in library &2.
Create Service Configuration (CRTSRVCFG)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Create Service Configuration (CRTSRVCFG) command creates the service configuration needed for all service and support applications: Electronic Customer Support (ECS), Electronic Service Agent, and Information Center Update.

Connectivity options are available from either local or remote systems or logical partitions. Primary or backup configurations can be created for the service configuration.

Restrictions:
• Input/output system configuration (*IOSYSCFG) special authority is required to run this command.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>ROLE</td>
<td>Role</td>
<td>*PRIMARY, *BACKUP</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td>CNTRYID</td>
<td>Country or region ID</td>
<td>Character value, *SELECT</td>
<td>Optional</td>
</tr>
<tr>
<td>STATE</td>
<td>State or province code</td>
<td>Character value, *SELECT</td>
<td>Optional</td>
</tr>
<tr>
<td>TELNBR1</td>
<td>Primary telephone number</td>
<td>Character value, *SELECT</td>
<td>Optional</td>
</tr>
<tr>
<td>TELNBR2</td>
<td>Alternate telephone number</td>
<td>Character value, *SELECT</td>
<td>Optional</td>
</tr>
<tr>
<td>RSRSCNAME</td>
<td>Resource name</td>
<td>Name, *CALC, *SELECT</td>
<td>Optional</td>
</tr>
<tr>
<td>MODM</td>
<td>Modem information name</td>
<td>Character value, *RSRCNAME, *SELECT</td>
<td>Optional</td>
</tr>
<tr>
<td>RMTSYS</td>
<td>Remote system</td>
<td>Character value</td>
<td>Optional</td>
</tr>
<tr>
<td>PROXY</td>
<td>Proxy server</td>
<td>Single values: *NO Other values: Element list</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Element 1: IP address or host name</td>
<td>Character value, *NONE</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Port number</td>
<td>1-65535, *IBMSVR</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 3: Relative priority</td>
<td>*TRYAFTER, *TRYBEFORE</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 4: Authentication user ID</td>
<td>Character value, *NONE</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 5: Authentication password</td>
<td>Character value, *NONE</td>
<td></td>
</tr>
<tr>
<td>ISPPRF</td>
<td>ISP profile name</td>
<td>Character value, *SELECT</td>
<td>Optional</td>
</tr>
</tbody>
</table>

© Copyright IBM Corp. 1998, 2006
<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNNPNT</td>
<td>Connectivity for others</td>
<td>Single values: *NO Other values: Element list</td>
<td>Optional</td>
</tr>
<tr>
<td>Element 1: Connection point</td>
<td>*YES</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Element 2: Interfaces</td>
<td>Values (up to 12 repetitions): Element list</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Element 1: Interface</td>
<td>Character value, *ALL, *SELECT</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Element 2: L2TP profile name</td>
<td>Character value, *GEN, *SELECT</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**CNNPNTPRX**

<table>
<thead>
<tr>
<th>Element 1: Connection point proxy</th>
<th>Element list</th>
<th>Optional</th>
</tr>
</thead>
<tbody>
<tr>
<td>Element 1: Port number</td>
<td>1-65535, *IBMSVR</td>
<td></td>
</tr>
<tr>
<td>Element 2: Authentication user ID</td>
<td>Character value, *NONE</td>
<td></td>
</tr>
<tr>
<td>Element 3: Authentication password</td>
<td>Character value, *NONE</td>
<td></td>
</tr>
</tbody>
</table>

---

**Role (ROLE)**

Specifies whether this service configuration is to be the primary or backup service configuration. The system or logical partition will first attempt to connect to IBM using the primary service configuration. If the primary service configuration fails, the system or logical partition will attempt the connection using a backup service configuration.

This is a required parameter.

*PRIMARY

The service configuration created will be the primary connection to IBM. A primary service configuration must be created before a backup service configuration can be created.

*BACKUP

The service configuration created will be the backup connection to IBM. This value is not allowed if the primary service configuration has not been created.

---

**Connection type (CNNTYPE)**

Specifies the connection type by which the system or logical partition will connect to IBM. You can connect to IBM using the current system or logical partition or through another system or logical partition.

This is a required parameter.

*DIRECT

Connect to IBM through the current system or logical partition using a direct connection to the internet as the connection type. Use this option if the system or logical partition can access the internet using one or more active TCP/IP interfaces.

*OTHERISP

Connect to IBM through the current system or logical partition using an internet service provider (ISP) as the connection type. Use this option if the system or logical partition can connect to an ISP using a point-to-point (PPP) connection profile.
*LCLDIAL
Connect to IBM through the current system or logical partition with a dial connection using AT&T Global Network Services (AGNS).

*MULTIHOP
Connect to IBM through another system or logical partition using a multi-hop connection to the internet as the connection type. Use this option if the system or logical partition providing the connection is configured to use CNNTYPE(*DIRECT), CNNTYPE(*OTHERISP), or CNNTYPE(*MULTIHOP).

*RMTDIAL
Connect to IBM through another system or logical partition using AT&T Global Network Services (AGNS) as the connection type. Use this option if the system or logical partition providing the connection is configured to dial to IBM using AT&T Global Network Services (AGNS).

Country or region ID (CNTRYID)
Specifies the country or region identifier used for the service configuration.

Note: This parameter is only valid when *PRIMARY is specified for the Role (ROLE) parameter.

*SELECT
A panel is displayed that allows the selection of a country or region identifier.

Note: This value is only valid if this command is run in an interactive job.

character-value
Specify the 2-character country or region identifier to be used.

State or province code (STATE)
Specifies the state or province code used for the service configuration.

Note: This parameter is only valid when *PRIMARY is specified for the Role (ROLE) parameter.

*SELECT
A panel is displayed that allows the selection of a state or province code. No selection panel is displayed if the specified country or region does not have states or provinces.

Note: This value is only valid if this command is run in an interactive job.

character-value
Specify the 2-character state or province code to be used.

Primary telephone number (TELNBR1)
Specifies the primary telephone number that will be dialed to connect to AT&T Global Network Services (AGNS).

Note: This parameter is only valid when *LCLDIAL is specified for the Connection type (CNNTYPE) parameter.
A panel is displayed that allows the selection of the primary telephone number. After a selection is made, an additional panel will be displayed to allow editing of the telephone number, adding any numbers or characters needed to obtain an outside line, pause while dialing, etc.

**character-value**
Specify the primary telephone number that will be dialed. Up to 48 characters can be specified.

### Alternate telephone number (TELNBR2)
Specifies the alternate telephone number that will be dialed to connect to AT&T Global Network Services (AGNS), if the connection attempt using the primary telephone number is unsuccessful.

**Note:** This parameter is only valid when *LCLDIAL is specified for the Connection type (CNNTYPE) parameter.

A panel is displayed that allows the selection of the alternate telephone number. After a selection is made, an additional panel will be displayed to allow editing of the telephone number, adding any numbers or characters needed to obtain an outside line, pause while dialing, etc.

**character-value**
Specify the alternate telephone number that will be dialed. Up to 48 characters can be specified.

### Resource name (RSRCNAME)
Specifies the communications resource that will be used by this service.

**Note:** This parameter is only valid when *LCLDIAL is specified for the Connection type (CNNTYPE) parameter.

**CALC**
The resource name will be determined as follows:
The internal communication resources that can use an integrated modem are determined. If only one integrated modem is defined, that resource will be used for the service configuration. The value *CALC is not valid if more than one integrated modem is defined.

If an integrated modem cannot be used, the resource cannot be calculated and it will have to be specified explicitly.

**SELECT**
A panel is displayed that allows the selection of the resource name that will be used.

**name**
Specify the name of the communications resource that will be used.

**Note:** Use the Work with Hardware Resources (WRKHDWRSC) command with *CMN specified for the TYPE parameter to help determine the resource name.

### Modem information name (MODEM)
Specifies the name of the modem description to use for this point-to-point service configuration.
Note: This parameter is only valid when *LCLDIAL is specified for the Connection type (CNNTYPE) parameter.

*RSRCNAME
The modem name will be determined based on the value specified for the Resource name (RSRCNAME) parameter. If the resource is defined to use an integrated modem, the appropriate internal modem description will be used. If the resource does not have a predefined modem description, MODEM(*RSRCNAME) cannot be used and the modem description must be specified explicitly.

*SELECT
A panel is displayed that allows the selection of the modem description that will be used.

character-value
Specify the name of the modem to use.

Note: The modem name must match one of the modems defined for the system or logical partition.

Remote system (RMTSYS)
Specifies either the IP address or host name (up to 255 characters) of the remote system or logical partition that will be used as the remote system that provides service configuration connectivity to IBM. A valid IP Version 4 address is accepted.

If the local system or logical partition has a service configuration created with CNNTYPE(*RMTDIAL) and has defined a remote system (RMTSYS) parameter, the remote system or logical partition must have a service configuration created with CNNTYPE(*LCLDIAL) and CNNPNT(*YES).

If the local system or logical partition has a service configuration created with CNNTYPE(*MULTIHOP) and has defined a remote system (RMTSYS) parameter, the remote system or logical partition must have a service configuration created with a connection type (CNNTYPE) of *DIRECT, *OTHERISP, or *MULTIHOP, and a connection point (CNNPNT) value of *YES.

character-value
Specify the IP address or host name of the remote system or logical partition that will provide the service configuration connection to IBM.

Proxy server (PROXY)
Specifies the information for configuring an HTTP or Service and Support proxy connection configuration. Configuring a proxy connection is optional. The proxy connection is an alternate connection configuration to the primary or backup connection that has been previously configured. A proxy connection can be used with any connection type.

The information provides the server with the information needed to connect to the destination proxy server.

Single values
*NO The IP address or host name is not specified. A proxy connection configuration will not be created.

Element 1: IP address or host name
*NONE
The IP address or host name is not specified. A proxy connection configuration will not be created.

character-value
Specify the IP address or host name of the proxy server through which this server will attempt to connect.

Element 2: Port number
*IBMSVR
The Service and Support proxy server will accept connections using the default port.

1-65535
Specify the port number on which the Service and Support proxy server will accept connections.

Element 3: Relative priority
*TRYAFTER
The proxy connection configuration will be attempted after the previously defined configuration.

*TRYBEFORE
The proxy connection configuration will be attempted before the previously defined configuration.

Element 4: Authentication user ID
*NONE
A user ID is not required.

character-value
If the proxy server requires authentication, specify the user ID that will be used.

Element 5: Authentication password
*NONE
A password is not required.

character-value
If the proxy server requires authentication, specify the password that will be used.

ISP profile name (ISPPRF)
Specifies the internet service provider (ISP) profile that will be used.

Note: This parameter is only valid when *OTHERISP is specified for the Connection type (CNNTYPE) parameter.

*SELECT
A panel is displayed that allows the selection of the ISP profile that will be used.

character-value
Specify the name of the ISP profile that will be used.
Connectivity for others (CNPNPNT)

Specifies whether other systems or logical partitions are allowed to use the service configuration connection to IBM through this system or logical partition.

**Note:** This parameter is only valid when *PRIMARY* is specified for the Role (ROLE) parameter.

**Single values**

*NO  Other systems or logical partitions are not allowed to use the service configuration connection to IBM configured on this system or logical partition.

**Element 1: Connection point**

*YES  Other systems or logical partitions are allowed to use the service configuration connection to IBM configured on this system or logical partition.

**Element 2: Interfaces**

Specifies which interfaces will listen for connections. Both the L2TP terminator profile and the Service and Support proxy, if configured, listen on the same interfaces. Up to 12 values can be specified.

**Element 1: Interface**

*ALL  All available interfaces will listen for a connection.

*SELECT  Only selected interfaces will listen for a connection.

**character-value**  Specify the interface that will listen for a connection.

**Element 2: L2TP profile name**

*GEN  Automatically generate and name an L2TP profile to use as the terminator profile.

*SELECT  Select an existing L2TP profile to use as the terminator profile.

**character-value**  Specify the name of an L2TP terminator profile which will be used to provide connectivity for other systems or logical partitions.

Connection point proxy (CNPNTPRX)

Specifies the Service and Support proxy server to provide connectivity for other systems or logical partitions.

**Element 1: Port number**

*IBMSVR  The Service and Support proxy server will accept connections using the default port.
Specify the port number on which the Service and Support proxy server will accept connections.

**Element 2: Authentication user ID**

*NONE

A user ID is not required.

**character-value**

If the proxy server requires authentication, specify the user ID that will be used.

**Element 3: Authentication password**

*NONE

A password is not required.

**character-value**

If the proxy server requires authentication, specify the password that will be used.

---

**Examples**

**Example 1: Creating a Primary Direct Service Configuration**

CRTSRVCFG  ROLE(*PRIMARY)  CNNTYPE(*DIRECT)
           CNTRYID(XX)  STATE(XX)

This command creates a primary direct internet service configuration used for all service and support applications: Electronic Customer Support (ECS), Electronic Service Agent, and Information Center Update.

**Example 2: Creating a Backup Service Configuration Using an Existing ISP**

CRTSRVCFG  ROLE(*BACKUP)  CNNTYPE(*OTHERISP)  CNTRYID(XX)
           STATE(XX)  ISPPRF(MYISP)  CNNPNT(*YES *ALL)

This command creates an backup internet service configuration using an internet service provider used for all service and support applications: Electronic Customer Support (ECS), Electronic Service Agent, and Information Center Update. The local system or logical partition is a connection point which will listen for connection requests on all interfaces.

**Example 3: Creating a Primary Local Dial Service Configuration**

CRTSRVCFG  ROLE(*PRIMARY)  CNNTYPE(*LCLDIAL)  CNTRYID(XX)
           STATE(XX)  TELNBR1(1111111)  TELNBR2(2222222)

This command creates a primary local dial connection to AT&T Global Network Services (AGNS) used for all service and support applications: Electronic Customer Support (ECS), Electronic Service Agent, and Information Center Update.

**Example 4: Creating a Backup Multi-hop Service Configuration**

CRTSRVCFG  ROLE(*BACKUP)  CNNTYPE(*MULTIHOP)  CNTRYID(YY)
           RMTSYS(ABCDEFG)

This command creates a backup internet service configuration using another remote system or logical partition used for all service and support applications: Electronic Customer Support (ECS), Electronic Service Agent, and Information Center Update. The configuration will connect using a multi-hop connection configuration through the remote system named ABCDEFG.
Example 5: Creating a Remote Point-to-Point Service Configuration

```plaintext
CRTSRVCFG  ROLE(*PRIMARY)  CNNTYPE(*RMTDIAL)  CNTRYID(XX)  
          STATE(XX)  RMTSYS(ABCDEFG)
```

This command creates a primary remote dial connection to AT&T Global Network Services (AGNS) used for all service and support applications: Electronic Customer Support (ECS), Electronic Service Agent, and Information Center Update. This configuration will connect using a dial connection configuration on a remote system named ABCDEFG.

Error messages

*ESCAPE Messages

CPFB040
If RSRCNAME(*SELECT) is specified, MODEM(*RSRCNAME) cannot be specified.

TCP8050
*IOSYSCFG authority required to use &1.

TCP8290
No TCP/IP point-to-point modem information

CPF8813
No entries exist.

CPF9899
Error occurred during processing of command.

TCP8205
Required object &2/&1 type *&3 not found.

TCP8211
Point-to-point profile &1 not found.
IBM Systems - iSeries: i5/OS Commands Starting with CRTMSGF (Create Message File)
Create Service Program (CRTSRVPGM)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Create Service Program (CRTSRVPGM) command creates a bound service program from a set of modules and binding directories.

Restrictions:
- You must have read (*READ) and add (*ADD) authorities to the library where the service program is to be created.
- You must have use (*USE) authority to the specified modules, service programs, and binding directories.
- You must have object operation (*OBJOPR) and *READ authorities to the file specified for the Source file (SRCFILE) parameter.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>SRVPGM</td>
<td>Service program</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Service program</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>MODULE</td>
<td>Module</td>
<td>Single values: *SRVPGM Qualified object name</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Module</td>
<td>Generic name, name, *ALL</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB, *USRLIBL</td>
<td></td>
</tr>
<tr>
<td>EXPORT</td>
<td>Export</td>
<td>*SRCFILE, *ALL</td>
<td>Optional</td>
</tr>
<tr>
<td>SRCFILE</td>
<td>Export source file</td>
<td>Qualified object name</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Export source file</td>
<td>Name, QSRVSRC</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td>Optional</td>
</tr>
<tr>
<td>SRCMBR</td>
<td>Export source member</td>
<td>Name, *SRVPGM</td>
<td>Optional</td>
</tr>
<tr>
<td>TEXT</td>
<td>Text ‘description’</td>
<td>Character value, *BLANK</td>
<td>Optional</td>
</tr>
<tr>
<td>BNDSRVPGM</td>
<td>Bind service program</td>
<td>Single values: *NONE Qualified object name</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Bind service program</td>
<td>Generic name, name, *ALL</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL</td>
<td></td>
</tr>
<tr>
<td>BNDDIR</td>
<td>Binding directory</td>
<td>Single values: *NONE Qualified object name</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Binding directory</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB, *USRLIBL</td>
<td>Optional</td>
</tr>
<tr>
<td>ACTGRP</td>
<td>Activation group</td>
<td>Name, *CALLER</td>
<td>Optional</td>
</tr>
<tr>
<td>Keyword</td>
<td>Description</td>
<td>Choices</td>
<td>Notes</td>
</tr>
<tr>
<td>-----------</td>
<td>---------------------------</td>
<td>-------------------------------------------------------------------------</td>
<td>---------</td>
</tr>
<tr>
<td>DETAIL</td>
<td>Listing detail</td>
<td>*NONE, *BASIC, *EXTENDED, *FULL</td>
<td>Optional</td>
</tr>
<tr>
<td>ALWUPD</td>
<td>Allow update</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
<tr>
<td>ALWLIBUPD</td>
<td>Allow *SRVPGM library update</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
<tr>
<td>USRPRF</td>
<td>User profile</td>
<td>*USER, *OWNER</td>
<td>Optional</td>
</tr>
<tr>
<td>REPLACE</td>
<td>Replace program</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
<tr>
<td>TGTRLS</td>
<td>Target release</td>
<td>Character value, *CURRENT, *PRV</td>
<td>Optional</td>
</tr>
<tr>
<td>ALVRINZ</td>
<td>Allow reinitialization</td>
<td>*YES</td>
<td>Optional</td>
</tr>
<tr>
<td>STGMDL</td>
<td>Storage model</td>
<td>*SNGLVL, *TERASPACE, *INHERIT</td>
<td>Optional</td>
</tr>
<tr>
<td>IPA</td>
<td>Interprocedural analysis</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
<tr>
<td>IPACTLFILE</td>
<td>IPA control file</td>
<td>Path name, *NONE</td>
<td>Optional</td>
</tr>
</tbody>
</table>

**Service program (SRVPGM)**

Specifies the service program object to be created.

This is a required parameter.

**Qualifier 1: Service program**

*name* Specify the name of the service program to be created.

**Qualifier 2: Library**

*CURRIB* The service program is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

*name* Specify the name of the library where the service program is to be created.

**Module (MODULE)**

Specifies the list of modules that are copied and bound together to create the service program object. If duplicate module and library specifications are found, only the first instance of the duplicate module and library is used. Modules in this list are copied into the final service program object. Up to 300 names can be specified.

**Single values**

*SRVPGM* The module and library names specified for the Service program (SRVPGM) parameter are used.

**Qualifier 1: Module**

*ALL* Find all module objects in the specified library or libraries.
**generic-name**
Specify all module objects starting with the characters preceding the * in the specified library or libraries.

**name** Specify the name of the module that is copied to create the service program object.

**Qualifier 2: Library**

**LIBL** All libraries in the library list for the current thread are searched until the first match is found.

**CURLIB** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

**USRLIBL** Only the libraries in the user portion of the job’s library list are searched.

**name** Specify the name of the library to be searched.

---

**Export (EXPORT)**

Specifies the names of the data and procedures this service program exports.

**SRCFILE**
The source file member identified by the Source file (SRCFILE) and Source member (SRCMBR) parameters contains EXPORT statements that identify the data and procedures to export from the service program.

**ALL** All data and procedures that are exported from the specified modules are also exported from the service program.

---

**Source file (SRCFILE)**

Specifies the source file containing the specifications for exporting data and procedures from this service program.

**Qualifier 1: Export source file**

**QSRVSRC** The source file containing the specifications for exporting data and procedures is named QSRVSRC.

**name** Specify the name of the source file containing the specifications for exporting data and procedures.

**Qualifier 2: Library**

**LIBL** All libraries in the library list for the current thread are searched until the first match is found.

**CURLIB** The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

**name** Specify the name of the library to be searched.
Source member (SRCMBR)

Specifies the name of the source file member containing the specifications for exporting data and procedures from this service program.

*SRVPGM

The source file member name is the same name as the service program name specified for the Service program (SRVPGM) parameter.

name Specify the name of the member in the source file containing the specifications for exporting data and procedures.

Text ’description’ (TEXT)

Specifies text that briefly describes the service program object.

*BLANK

Text is not specified.

character-value

Specify no more than 50 characters of text, enclosed in apostrophes.

Bind service program (BNDSRVPGM)

Specifies the list of service program exports to examine at bind time to ensure they satisfy any module import requests. The service program exports are checked only if there are unresolved module import requests not satisfied by the set of module exports. Any service program specified on the BNDSRVPGM parameter that satisfies a module import request will be bound to the service program being created. The service program name and the library specified on the BNDSRVPGM parameter are saved to be used at run time. Up to 300 names can be specified.

Single values

*NONE

No service programs are provided for symbol resolution.

Qualifier 1: Bind service program

*ALL Find all service program objects in the specified library or libraries.

Note: This value should only be specified in a user-controlled environment when you know exactly what is getting bound to your service program. Specifying *BLIB with *ALL may give you unpredictable results at service program run time. Specify the generic service program name or specific libraries to better control what gets bound to your service program.

generic-name

Specify all service program objects starting with the characters preceding the * in the specified library or libraries.

name Specify the name of the service program to be examined during symbol resolution.

Qualifier 2: Library

*BLIB Search all of the libraries in the job’s library list for the specified service programs. If one of the service programs has an export that satisfies a module import, then the library list is searched at run-time to find this service program.
name Specify the name of the library where the service programs can be found. If one of the service programs has an export that satisfies a module import, and a specific library was specified, the specified library is searched at run-time to find this service program.

Note: QTEMP is not a valid library name for this parameter.

**Binding directory (BNDDIR)**

Specifies the list of binding directories that are used in symbol resolution. Up to 300 names can be specified.

**Single values**

*NONE No binding directory is specified.

Qualifier 1: Binding directory

name Specify the name of the binding directory used in symbol resolution.

Qualifier 2: Library

*LIBL All libraries in the library list for the current thread are searched until the first match is found.

*CURLIB

The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

*USRLIBL

Only the libraries in the user portion of the job’s library list are searched.

name Specify the name of the library to be searched.

**Activation group (ACTGRP)**

Specifies the activation group this service program is associated with when it is called. An activation group provides:

- Run-time data structures to support the running of programs
- Addressing protection
- A logical boundary for message creation
- A logical boundary for application cleanup processing

*CALLER

When this service program gets called, the service program is activated into the caller’s activation group.

name Specify the name of the group that is associated with this called service program. If an activation group, by the specified name, currently exists when this service program is called, the service program is associated with the already existing activation group. If an activation group, by the specified name, does not currently exist when this service program is called, then a new activation group is created and the service program is associated with the newly created activation group.
Creation options (OPTION)

Specifies options to be used when the service program object is created.

You can specify up to 5 values for this parameter.

Service Program Objects

*GEN     A service program object is generated.

*NOSAPOOL     A service program object is not generated.

Duplicate Procedure Names

*NODUPPROC     During the symbol resolution phase of the binding process, each procedure name that is exported from the modules and service programs must be unique.

*DUPPROC     During the symbol resolution phase of the binding process, the procedure names that are exported from the modules and service programs do not have to be unique. When multiple duplicate procedures are allowed, the first exported procedure in the list of specified modules and service program that matches the import request is the procedure that is selected.

Duplicate Variable Names

*NODUPVAR     During the symbol resolution phase of the binding process, each variable name that is exported from the modules and service programs must be unique.

*DUPVAR     During the symbol resolution phase of the binding process, the variable names that are exported from the modules and service programs do not have to be unique. When multiple duplicate variables are allowed, the first exported variable in the list of specified modules and service programs that matches the import request is the variable that is selected.

Issuing Diagnostic Messages

*WARN     If duplicate variables or procedures are found, then a diagnostic message is issued indicating what duplicates were found.

*NOWARN     If duplicate variables or procedures are found, diagnostic messages are not issued.

Resolving References (Imports)

*RSLVREF     All imports must be resolved to exports for the service program to be created.

*UNRSLVREF     All imports do not need to resolve to exports for the service program to be created. If the service program tries to use one of these unresolved imports at run time, a MCH4439 run-time exception is issued.
Listing detail (DETAIL)

Specifies the level of detail to be printed.

*NONE

A listing is not generated.

*BASIC

Contains a listing of the options passed to CRTPGM, and processing statistics. This listing also contains the Brief Summary Table.

*EXTENDED

In addition to the information provided in the *BASIC listing, this listing contains the Extended Summary Table and the Binding Information Listing.

*FULL

This listing contains the *EXTENDED listing and the Cross-Reference Listing.

Note: If a printed listing is requested, the printer file *LIBL/QSYSPRT is used to generate the listing.

Allow update (ALWUPD)

Specifies whether to allow an update of the service program being created using the Update Service Program (UPDSRVPGM) command.

*YES

The service program can be updated using the UPDSRVPGM command.

*NO

The UPDSRVPGM command cannot be used to update the service program being created.

Allow *SRVPGM library update (ALWLIBUPD)

Specifies whether to allow the bound service program library name of the service program being created to be changed when updated using the UPDSRVPGM command.

*NO

The UPDSRVPGM command is not allowed to update the bound service program library names of the service program being created, even if *YES is specified for the Allow update (ALWUPD) parameter.

*YES

The UPDSRVPGM command is allowed to update the bound service program library names of the service program being created when ALWUPD(*YES) is specified.

User profile (USRPRF)

Specifies whether authority checking is performed only for the user running the service program, or for both the user running the service program and the service program owner.

*USER

The user profile of the service program user is used when the service program is run.

*OWNER

The user profile of both the service program owner and the service program user is used when the service program is run.
Replace program (REPLACE)

Specifies whether the existing service program is replaced if a service program by the same name already exists in the specified library.

*YES  Replace the existing service program by moving it to the QRPLOBJ library. Current activations of the service program will continue running using the version of the service program in the QRPLOBJ library.

Note: Both service programs must be owned by the same user for the replace to work.

*NO   No replacement occurs. An error message is issued if a service program already exists with the name and library specified for the &Service program (SRVPGM) parameter.

Authority (AUT)

Specifies the authority you are giving to users who do not have specific authority for the object, who are not on an authorization list, and whose group profile or supplemental group profiles do not have specific authority for the object.

*LIBCRTAUT

The system determines the authority for the object by using the value specified for the Create authority (CRTAUT) parameter on the Create Library command (CRTLIBM) for the library containing the object to be created. If the value specified for the CRTAUT parameter is changed, the new value will not affect any existing objects.

*CHANGE

The user can perform all operations on the object except those limited to the owner or controlled by object existence (*OBJEXIST) and object management (*OBJMGT) authorities. The user can change and perform basic functions on the object. *CHANGE authority provides object operational (*OBJOPR) authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove users.

*ALL   The user can perform all operations except those limited to the owner or controlled by authorization list management (*AUTLMGT) authority. The user can control the object’s existence, specify the security for the object, change the object, and perform basic functions on the object. The user also can change ownership of the object.

*USE   The user can perform basic operations on the object, such as running a program or reading a file. The user cannot change the object. Use (*USE) authority provides object operational (*OBJOPR), read (*READ), and execute (*EXECUTE) authorities.

*EXCLUDE

The user cannot access the object.

name  Specify the name of an authorization list to be used for authority to the object. Users included in the authorization list are granted authority to the object as specified in the list. The authorization list must exist when the object is created.

Target release (TGTRLS)

Specifies the release of the operating system on which you intend to use the object being created.
When specifying the **target-release** value, the format VxRxMx is used to specify the release, where Vx is the version, Rx is the release, and Mx is the modification level. For example, V5R3M0 is version 5, release 3, modification 0.

Valid values depend on the current version, release, and modification level of the operating system, and they change with each new release. You can press F4 while prompting this command parameter to see a list of valid target release values.

**CURRENT**
The object is to be used on the release of the operating system currently running on your system. The object can also be used on a system with any subsequent release of the operating system installed.

**PRV**
The object is to be used on the previous release with modification level 0 of the operating system. The object can also be used on a system with any subsequent release of the operating system installed.

**character-value**
Specify the release in the format VxRxMx. The object can be used on a system with the specified release or with any subsequent release of the operating system installed.

---

**Allow reinitialization (ALWRINZ)**
Specifies if the static storage of the service program is allowed to be reinitialized while it is still active.

**NO**
The static storage of the service program can not be reinitialized while it is still active.

**YES**
The static storage of the service program is allowed to be reinitialized while the service program is still active.

---

**Storage model (STGMDL)**
Specifies the storage model attribute of the service program.

**SNGLVL**
The service program is created with single-level storage model. When a single-level storage model service program is activated and run, it is supplied single-level storage for automatic and static storage. A single-level storage service program runs only in a single-level storage activation group.

**TERASPACE**
The service program is created with teraspace storage model. When a teraspace storage model service program is activated and run, it is supplied teraspace storage for automatic and static storage. A teraspace storage service program runs only in a teraspace storage activation group.

**INHERIT**
The service program is created with inherit storage model. When activated, the service program adopts the storage model of the activation group into which it is activated. An equivalent view is that it inherits the storage model of its caller. When the *INHERIT* storage model is selected, *CALLER* must be specified for the **Activation group (ACTGRP)** parameter.
Interprocedural analysis (IPA)

Specifies whether interprocedural analysis (IPA) is to be used during the service program creation. For more information on IPA, refer to the ILE Concepts book, SC41-5606.

*NO  Interprocedural analysis will not be performed.
*YES  Interprocedural analysis will be performed.

IPA control file (IPACTLFILE)

Gives the path name of a file which contains interprocedural analysis (IPA) suboption information. This parameter is allowed only when IPA(*YES) is specified.

*NONE  No IPA control file information is to be used when IPA(*YES) is specified.

path-name  Specify the path name of the IPA control file to use when IPA(*YES) is specified. If the name is qualified it must be enclosed in apostrophes. An example of a qualified IPA control file name is '/directory1/directory2/myipactlfname'

Examples

CRTSRVPGM  SRVPGM(WORKDOC)

This command creates the service program object named WORKDOC in the current library. The service program will be created from one module object that is also named WORKDOC and is located using the current library for the job.

Error messages

*ESCAPE Messages

CPF223E  Authority check for use adopted authority attribute failed.

CPF5D05  Service program &1 not created.

CPF5D07  Export source file record length greater than 240.

CPF5D12  Error encountered during program or service program preparation.
Create Tape Category (CRTTAPCGY)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Create Tape Category (CRTTAPCGY) command creates a user defined category name and assigns it to a system name. A category may be used to "group" volume identifiers together. The following special value categories have been provided by the operating system and are defined for all library devices.

- **NOSHARE**
  For D/T3494 tape media libraries the volume identifier may only be used by the system that owns the rights of the *NOSHARE. Other tape media libraries can have cartridge identifiers in the *NOSHARE category, but the security of the *NOSHARE is not handled by the operating system.

- **SHARE400**
  The cartridge identifier may be shared by all systems that are attached to the library device.

- **IPL**
  This category should be used for cartridge identifiers that would be used in an alternate IPL. The management of the cartridges in the category must be done by the user.

- **NL**
  The cartridge identifier in this category must have no logical volume identifier in order for it to be used. If it has a logical volume identifier, an Initialize Tape (INZTAP) command must be done to initialize the tape volume to a non-labeled tape before it can be used.

- **INSERT**
  The cartridge identifier has been placed in the library device, but has not yet been added to the system. An Add Tape Cartridge (ADDTAPCTG) command must be done before any I/O may be done to the tape volume.

- **EJECT**
  The volume identifier has been removed from the system by a Remove Tape Cartridge (RMVTAPCTG) command and is no longer usable by the library device.

- **CNV**
  When a tape in this category is unloaded by specifying ENDOPT(*UNLOAD), OS/400 will automatically export the tape to the convenience station.

- **SYSGEN**
  The cartridge ID exists in the *SYSGEN category. *SYSGEN category is used for all cartridges when the library device description is in *SYSGEN mode. A library device description is in *SYSGEN mode when the GENCTGID parameter is *SYSGEN in the device description. *SYSGEN mode is used to generate identifiers for non-bar code libraries instead of loading and reading all the logical volume identifiers directly from the tape. If the library device is in *SYSGEN mode, cartridges cannot be moved from the *SYSGEN category.
Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>CGY</td>
<td>Category</td>
<td>Element list</td>
<td>Required,</td>
</tr>
<tr>
<td></td>
<td>Element 1: Category name</td>
<td>Character value</td>
<td>Positional 1</td>
</tr>
<tr>
<td></td>
<td>Element 2: Category system</td>
<td>Character value</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>*CURRENT</td>
</tr>
</tbody>
</table>

Category (CGY)

 Specifies the category being created.

**Element 1: Category name**

*character-value*

Specify the name of the category to create.

**Element 2: Category system**

Identifies the system the category belongs to. The system name is obtained from the current system name field of a Display Network Attributes (DSPNETA) command.

*CURRENT

The system currently running the command.

*character-value*

Specify the name of the system that the category belongs to. Do not attempt to create a category and specify an owning system other than *CURRENT, unless the category has previously been defined on the system specified. For example, If system A and system B are attached to library device LIB01, then CRTAPCGY CGY(CAT1 A) creating category CAT1 and assigning as its owner system A is required on system A before a CRTAPCGY CGY(CAT1 A) creating category CAT1 and assigning as its owner system A can be done on system B. If both of these create commands are successful, CAT1 owned by system A is logically considered the same category and can be used for cartridges in library LIB01.

**Examples**

CRTPAPCGY  CGY(CAT1 RCHAS215)

This command creates a user defined category named CAT1 and assigns as its primary owner system RCHAS215.

**Error messages**

*ESCAPE Messages*

CPF67DD

Category not created.
CPF67E2
Category already exists
Create Tape File (CRTTAPF)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Create Tape File (CRTTAPF) command creates a tape device file. The device file contains the file description, which identifies the device to be used; it does not contain data. The tape device file is used to read and write records on tape. The same device file can be used for both input and output operations.

Note: This command is not used to create device files for use in save or restore operations. User-created device files are not needed for save or restore operations.

Tape files have no data description specifications (DDS). The information in the tape file description comes from the command that creates it. The tape file has one record format for input/output operations. The record format consists of one character field containing the input data retrieved from the device or the output data to be written to the device. The program using the device file must describe the fields in the record format so the program can arrange the data received from or sent to the device in the manner specified by the tape file description.

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FILE</strong></td>
<td>File</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: File</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *CURLIB</td>
<td></td>
</tr>
<tr>
<td><strong>DEV</strong></td>
<td>Tape device</td>
<td>Single values: *NONE Other values (up to 4 repetitions): Name</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>VOL</strong></td>
<td>Volume identifier</td>
<td>Single values: *NONE Other values (up to 50 repetitions): Character value</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>REELS</strong></td>
<td>Tape reels specifications</td>
<td>Element list</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Element 2: Number of reels</td>
<td>1-255, 1</td>
<td></td>
</tr>
<tr>
<td><strong>SEQNBR</strong></td>
<td>Sequence number</td>
<td>1-16777215, 1, *END, *NEXT</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>LABEL</strong></td>
<td>Tape label</td>
<td>Character value, *NONE</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>FILETYPE</strong></td>
<td>File type</td>
<td>*DATA, *SRC</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>IGCDDTA</strong></td>
<td>User specified DBCS data</td>
<td>*NO, *YES</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>TEXT</strong></td>
<td>Text 'description'</td>
<td>Character value, *BLANK</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>RCDDLEN</strong></td>
<td>Record length</td>
<td>Integer, *CALC</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>BLKLEN</strong></td>
<td>Block length</td>
<td>1-524288, *CALC</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>BUFOSET</strong></td>
<td>Buffer offset</td>
<td>Integer, 0, *BLKDSC</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>EXTEND</strong></td>
<td>Extend</td>
<td>Single values: *NO Other values: Element list</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Element 1: Extend file</td>
<td>*YES</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Check file</td>
<td>*NOCHECK, *CHECK</td>
<td></td>
</tr>
<tr>
<td>Keyword</td>
<td>Description</td>
<td>Choices</td>
<td>Notes</td>
</tr>
<tr>
<td>---------</td>
<td>-------------</td>
<td>----------------------------------------------------------------------------</td>
<td>--------</td>
</tr>
<tr>
<td><strong>COMPACT</strong></td>
<td>Data compaction</td>
<td>*DEVD, *NO</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>CODE</strong></td>
<td>Code</td>
<td>*EBCDIC, *ASCII</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>CRTDATE</strong></td>
<td>Creation date</td>
<td>Date, *NONE</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>EXPDATE</strong></td>
<td>File expiration date</td>
<td>Date, *NONE, *PERM</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>ENDOPT</strong></td>
<td>End of tape option</td>
<td>*REWINd, *LEAVE, *UNLOADd</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>USRLBLPGM</strong></td>
<td>User label program</td>
<td>Single values: *NONEd&lt;br&gt;Other values: Qualified object name</td>
<td>Optional</td>
</tr>
<tr>
<td>Qualifier 1: User label program</td>
<td>Name</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qualifier 2: Library</td>
<td>Name, *LIBLd, *CURLIBd</td>
<td></td>
<td></td>
</tr>
<tr>
<td><strong>WAITFILE</strong></td>
<td>Maximum file wait time</td>
<td>Integer, *IMMED, *CLS</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>SHARE</strong></td>
<td>Share open data path</td>
<td>*NO, *YES</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>AUT</strong></td>
<td>Authority</td>
<td>Name, *LIBCRTAUTd, *ALLd, *CHANGEd, *EXCLUDEd, *USEd</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>REPLACE</strong></td>
<td>Replace file</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
</tbody>
</table>

---

**File (FILE)**

Specifies the tape device file to be created.

If the file is used by a high-level language program, the file name must be consistent with the naming rules of that language. Otherwise, the file must be renamed in the program itself.

This is a required parameter.

**Qualifier 1: File**

*name*  Specify the name of the tape device file to be created.

**Qualifier 2: Library**

* CURLIB  
  The current library for the job is used to locate the tape device file. If no library is specified as the current library for the job, QGPL is used.

*name*  Specify the library where the tape file is located.

---

**Device (DEV)**

Specifies the names of one or more tape devices, one virtual tape device, or one media library device used with this tape device file to perform reading and writing data operations. A media library device is a tape storage device that contains one or more tape drives, tape cartridges, and a part (carriage and picker assembly) for moving tape media between the cartridge storage slots and the tape drives.
Single values

*NONE
No device names are specified. They must be specified later in the Change Tape File (CHGTAPF) or Override Tape File (OVRTAPF) command, or in the high-level language program that opens the file.

Other values

name Specify the names of no more than four tape devices, one virtual tape device, or the name of one media library device used with this tape device file. The order in which the device names are specified here is the order in which tapes on those devices are processed. When the number of volumes being processed exceeds the number of devices listed on this parameter, the devices are used in the same order as specified, wrapping around to the first device as needed.

Volume identifier (VOL)

Specifies one or more volume identifiers used by the file. The volumes must be installed in the same order as the identifiers are specified here (and as they are specified for the DEV parameter. If the file is opened for read backward, then the volume identifiers in the list are processed from last to first (while the devices in the device list are used in first-to-last order). If a list of volume identifiers is provided for the file, operator messages indicate the name of the required volume.

Single values

*NONE
No tape volume identifiers are specified for this file. They can be supplied before the device file is opened, either in a CHGTAPF or OVRTAPF command or in the high-level language program. If volume identifiers are not specified before the device file is opened, volume checking is not performed beyond verifying that the correct label type volume is on the device, and volume names are not provided in operator messages. The maximum number of reels processed for an *NL, *NS, *BLP, or *LTM input file when VOL(*NONE) is specified is determined by the REELS(number-of-reels) parameter value.

Other values (up to 50 repetitions)

character-value
Specify the identifiers of one or more volumes in the order in which they are placed on the device. Each volume identifier contains a maximum of 6 alphanumeric characters. Use a blank as a separator character when listing multiple identifiers. Up to 50 volume identifiers can be specified. These identifiers are used in messages sent to the operator during processing. The maximum number of reels processed for an *NL, *NS, *BLP, or *LTM input file is determined by the number of volume identifiers in the list.

Note: If the VOL parameter value used for the file specifies a list of identifiers rather than VOL(*NONE), the number-of-reels part of the REELS parameter is ignored regardless of where it is specified. A description of how the parameter values for the file are determined when overrides are used, the high-level language interface, and the device file when the file is opened is in the Files and file systems topic in the iSeries Information Center at http://www.ibm.com/eserver/iseries/infocenter. To ensure that the number-of-reels part of the REELS parameter is used (rather than a VOL identifier list) to control the volumes processed by the tape device file, specify VOL(*NONE) in the same command in which the REELS parameter is specified.
Tape reels specifications (REELS)

Specifies the type of labeling used on the tape reels and the maximum number of reels processed if both a list of volume identifiers is not specified (VOL parameter) and this device file is used with either *NL, *NS, *LTM, or *BLP input files. When the number of reels is specified as the second element of this parameter, the volume identifiers on the volumes are ignored if labeled tapes are being processed; instead, the order in which the reels are installed on the device must be checked by the operator.

The number-of-reels value is not a limiting value for standard-label or output files. For a standard-label input file, the data file labels limit the number of volumes processed by indicating end-of-file. For an output file, the number-of-reels value is ignored; the system requests that additional volumes be kept on the device until the file is closed.

The system checks the first record following the load point on the tape to see (1) whether it has exactly 80 bytes for EBCDIC or at least 80 bytes for ASCII and (2) whether the first 4 bytes contain the values VOL and 1. If so, the reel contains a standard-label tape. *SL and *BLP files require standard-label tape volumes. *NL, *NS, and *LTM tape files cannot process standard-label volumes.

Note: The values *SL, *NL, and *LTM can be specified if the device file is used for either reading or writing on tapes. The values *NS and *BLP are valid only if the device file is used to read tapes.

Element 1: Label processing type

*SL The volumes have standard labels. If a list of volume identifiers is specified (with the VOL parameter), the system checks that the correct tape volumes are on the device in the specified sequence.

- If no volume identifier list is given and the file is opened for output, any standard-label volumes may be installed on the device.
- If no volume identifier list is given and the file is opened for input, the first volume may have any volume identifier, but if the file is continued, the system requires the correct continuation volumes to be processed (verified by checking the data file labels). For an input file, the end-of-file message is sent to the program being used when the labels on the last volume processed indicate that it is the last volume for the data file.

*NL The volumes are not labeled. On a nonlabeled volume, tape marks are used to indicate the end of each data file and the end of the volume. For an input file, the end-of-file message is sent to the program when the number of volumes specified in the volume list have been processed, or, if no list of volume identifiers is provided, when the number of reels specified in the REELS parameter are processed.

*NS The volumes have nonstandard labels. Each volume must start with some kind of label information, optionally preceded by a tape marker and always followed by a tape marker. This nonstandard label information is ignored. The system spaces forward to a point beyond the tape marker that follows the nonstandard labels and positions the tape at the file’s data. Each reel must have a tape marker at the end of the file’s data. Information beyond this ending tape marker is ignored. Only a single data file can exist on a nonstandard tape. Standard-label volumes cannot be processed by using the *NS label processing.

For an input file, the end-of-file message is sent to the program using the file when the number of volumes specified in the volume list have been processed, or, if no list of volume identifiers is provided, when the number of reels specified in the REELS parameter are processed.

*BLP Standard-label processing is bypassed. Each reel must have standard labels. Although each reel is checked for a standard volume label and each file must have at least one standard header label (HDR1) and one standard trailer label (EOF1 or EOF1), most other label information (such as the data file record length or block length) is ignored. The sequence number of each file on the
volume is determined only by the number of tape markers between it and the start of tape (in contrast to *SL processing in which the file sequence number stored in the header and trailer labels of each file are used to locate a data file).

Most of the information in the data file trailer label is ignored, but if an end-of-file (EOF) trailer label is found, the end-of-file message is sent to the program using the tape file. If no end-of-file trailer label is encountered by the time the specified number of volumes or reels have been processed (volume identifier list and REELS parameter), the end-of-file message is immediately sent to the program using the tape file. Bypass label processing can be used when the user does not know the name of the file used or when some file label information is incorrect.

*LTM The volumes have no labels but do have a single leading tape marker before the first data file. REELS(*LTM) is processed the same as REELS(*NL) except that when SEQNBR(1) is specified for an output file to create the first data file on the tape, a leading tape marker is written at the start of the tape before the first data block.

Element 2: Number of reels

1 Only one reel is processed for the *NL, *LTM, *NS, or *BLP tape file input operation if no list of volume identifiers is provided (VOL parameter).

1-255 Specify the maximum number of reels to be processed for an *NL, *LTM, *NS, or *BLP input tape operation when a list of volume identifiers is not specified (VOL parameter). If the next reel is not on the device when the end of the currently-processing tape is reached, a message is sent to the operator requesting that the next tape be installed on the next tape device. The number-of-reels value is ignored for a standard-label (SL) file or for any output file.

Sequence number (SEQNBR)

Specifies the sequence number of the data file on the tape being processed.

• When standard-label tapes are used, the four-position file sequence number is read from the first header label of the data file.

• When bypass label processing is used or when standard-label tapes are not used, the system counts the tape markers from the start of the tape to locate the correct sequence number data file to be processed.

• When multiple-file, multiple-volume tapes are processed using REELS(*SL), the file sequence numbers continue consecutively through the volumes; thus, each new data file has a sequence number one greater than the previous file, regardless of its volume location.

1 For standard-label tapes (not using bypass label processing), the data file having the sequence number 1 is processed. For nonlabeled tapes and for bypass label processing of standard-label tapes, the first data file on the tape is processed.

*END The file is written on the end of the tape. This value is used only for files that are written to tape. An error message is shown on the display when a tape device file is used to read from a tape and the *END special value is specified in the tape device file.

*NEXT The next file in the sequence is processed. This value is used for files read from tape. If the tape is currently in a position that is prior to the first file, the first file on the tape is processed.

An error message is shown on the display when a tape file is used to write to a tape and the *NEXT special value is specified in the tape file.

1-16777215 Specify the sequence number of the file.
Tape label (LABEL)

Specifies the data file identifier of the data file processed by this tape device file. An identifier is defined only for standard-label tapes and is stored in the header label immediately before the data file.

If a data file identifier is specified for any type of label processing other than *SL, it is ignored.

An identifier is required for a standard label output file, but is optional for an input file because the sequence number uniquely identifies the data file to process.

For an input file or output file with EXTEND(*YES) specified, this parameter specifies the identifier of the data file on the tape. The specified identifier must match the one in the labels of the data file that the SEQNBR parameter specifies; otherwise, an error message is sent to the program using this device file. For output files with EXTEND(*NO) specified, this parameter specifies the identifier of the data file to be created on the tape.

*NONE

The data file identifier is not specified.

character-value

Specify the identifier (17 alphanumeric characters maximum) of the data file used with this tape device file. If this identifier is for a tape written in the basic exchange format, and is used on a system other than an iSeries system, up to eight characters or a qualified identifier having no more than eight characters per qualifier must be used.

File type (FILETYPE)

Specifies whether the tape device file being created describes data records or source records (statements) for a program or for another file.

*DATA

The tape file describes data records.

*SRC

The tape file describes source records.

Note: If *SRC is specified, the system adds 12 bytes to the start of every record to replace the source sequence number and date fields.

User specified DBCS data (IGCDTA)

Specifies, for program-described files, whether the file processes double-byte character set (DBCS) data. Specifies, for externally described files, the DBCS attributes of the file.

*NO

The file does not process double-byte character set (DBCS) data.

*YES

The file processes double-byte character set (DBCS) data.

Text 'description' (TEXT)

Specifies the text that briefly describes the object.
**Record length (RCDLEN)**

Specifies, in bytes, the length of the records contained in the data file processed with this device file. The system always uses the record length and block length specified in the data file labels for any standard-label input file or output file with EXTEND(*YES) specified (if a second header label (HDR2) is found on the tape and *BLP label processing has not been specified).

**CALC**

No record length is specified for the data file being processed. If *CALC is specified, the system will attempt to calculate an appropriate record length when the file is opened. RCDLEN(*CALC) can be used for nonlabeled tapes or when there is no HDR2 label if a BLKLEN value other than *CALC is specified for the file and RCDBLKFM does not specify spanned or blocked records. In this case, the system calculates an appropriate record length from the block length, record block format, and buffer offset (for an ASCII file) specified for the file. In any other case, the actual record length must be specified by a CHGTAPF command or OVRTAPF command, or in the high-level language program that opens the device file. The system attempts to calculate an appropriate record length when the file is opened.

**integer**

Specify the length of each record in the data file. Valid values range from 1 through 32767 bytes.

### Table 1. Figure: EBCDIC RCDLEN Ranges

<table>
<thead>
<tr>
<th>RCDFBLKFMT</th>
<th>FILETYPE(+DATA)</th>
<th>FILETYPE(+SRC)</th>
</tr>
</thead>
<tbody>
<tr>
<td>*F +FB +U</td>
<td>10 - 32767</td>
<td>30 - 32767</td>
</tr>
<tr>
<td>*V +VB</td>
<td>1 - 32759</td>
<td>13 - 32767</td>
</tr>
<tr>
<td>*VS +VBS</td>
<td>1 - 32759</td>
<td>13 - 32767</td>
</tr>
</tbody>
</table>

### Table 2. Figure: ASCII RCDLEN Ranges

<table>
<thead>
<tr>
<th>RCDFBLKFMT</th>
<th>FILETYPE(+DATA)</th>
<th>FILETYPE(+SRC)</th>
</tr>
</thead>
<tbody>
<tr>
<td>+F +FB +U</td>
<td>18 - 32767</td>
<td>30 - 32767</td>
</tr>
<tr>
<td>+D +DB</td>
<td>1 - 9995</td>
<td>13 - 10007</td>
</tr>
<tr>
<td>+VS +VBS</td>
<td>1 - 32759</td>
<td>13 - 32767</td>
</tr>
</tbody>
</table>

**Block length (BLKLEN)**

Specifies the number of bytes in the maximum length of the data blocks being transferred to or from the tape for reading or writing operations.

**CALC**

No block length is specified for the data file being processed. The system attempts to calculate an appropriate block length when the file is opened.
Specify the maximum length of each block in the data file to be processed. The minimum block length that can be successfully processing is determined by the tape device hardware and iSeries system machine support functions.

The maximum block length is always 524288 bytes for an input file, but is limited to 9999 bytes if block descriptors must be created for an ASCII output file.

The following table shows the minimum and maximum block length values allowed for an output file:

<table>
<thead>
<tr>
<th>CODE</th>
<th>BUOFSET</th>
<th>MIN BLKLEN</th>
<th>MAX BLKLEN</th>
</tr>
</thead>
<tbody>
<tr>
<td>*EBCDIC</td>
<td>Ignored</td>
<td>18</td>
<td>524288</td>
</tr>
<tr>
<td>*ASCII</td>
<td>0</td>
<td>18</td>
<td>524288</td>
</tr>
<tr>
<td>*ASCII</td>
<td>+BLKSC</td>
<td>18</td>
<td>9999</td>
</tr>
</tbody>
</table>

**Buffer offset (BUOFSET)**

Specifies the buffer offset value for the start of the first record in each block in the tape data file. A buffer offset value can be used for any record block format ASCII file, and is ignored for an EBCDIC tape file.

The system uses the buffer offset specified in the data file labels for any standard-label input file or output file with EXTEND(*YES) specified if a value is contained in the second header label (HDR2) on the tape, and *BLP label processing has not been specified.

The buffer offset parameter specifies the length of any information that precedes the first record in the block. For record block formats *D, *DB, *VS, and *VBS, each record or record segment is preceded by a descriptor that contains the length of the record or segment. A buffer offset value is used to indicate that there is information ahead of the descriptor word for the first record in each block, or ahead of the data of the first fixed-length record or undefined format record in each block.

This parameter is not needed for a standard-label file processed for input if the tape includes a second file header label (HDR2) that contains the buffer offset value. A buffer offset value must be provided by the Create Tape File (CRTTAPF) command, Change Tape File (CHGTAPF) command, or Override Tape File (OVRTAPF) command, or by the file labels for an input file that contains any information (such as a block descriptor) ahead of the first record in each block. If the user does not specify a buffer offset value when a tape file is created, it is not necessary to specify an offset value when the file is read.

The only buffer offset values allowed for an output file are zero and *BLKSC. An existing standard-label data file with a buffer offset value in the HDR2 label can be extended only if the buffer offset value is either 0 or 4. A buffer offset value of 0 in the HDR2 label adds data blocks with no buffer offset. BUOFSET(*BLKSC) must be specified to extend an existing tape data file that contains an offset value of 4 in the HDR2 label.

- **0** No buffer offset information precedes the first record in each data block.
- **BLKSC** 4-byte block descriptors are created in any tape file created by using this device file. Any input file read by using this device file assumes 4-bytes of buffer offset information preceding the first record in each data block. This value is valid only if *D or *DB is specified for the Record block format (RCDBLKFMT) parameter.
integer

Specify the length (in bytes) of the buffer offset information that precedes the first record in each data block. Valid values range from 0 through 99 bytes.

---

Record block format (RCDBLKfMT)

Specifies the type and blocking attribute of records in the tape data file being processed.

Record block format *V and *VB records can be processed only for an EBCDIC file; *D and *DB records can be processed only for an ASCII file. If a standard-label tape (label type *SL or *BLP) is being processed and an inconsistent record block format is specified for the volume code, the correct record type is assumed (V or D) for the volume code and a warning message is sent to the program that opens the file. If the record type and code are inconsistent for a nonlabeled volume (label type *NL, *LTM, or *NS), an error message is sent and the file is not opened, because there are no labels to verify the correct volume code.

If a valid record length, block length, and buffer offset value (for an ASCII file) are specified for fixed-length records but the block attribute is incorrect, the correct block attribute is assumed (changing record block format *F to *FB or record block format *FB to *F), and a warning message is sent to the program that opens the file.

If a block length is specified that is longer than required to process a maximum length record, then record block format *V, *D, or *VS is changed to *VB, *DB, or *VBS and a warning message is sent to the program that opens the file.

Note: When BUFOFSET(*BLKIDC) is specified for the file, a value of 4 should be used for the BUOFSET part of any BLKLEN calculations, unless existing file labels on the tape specify a different value.

*FB Fixed length, blocked, unspanned records in either EBCDIC or ASCII code are processed.

*F Fixed length, deblocked, unspanned records in either EBCDIC or ASCII code are processed.

*V Variable length, deblocked, unspanned records in EBCDIC type V format are processed.

*VB Variable length, blocked, unspanned records in EBCDIC type V format are processed.

*D Variable length, deblocked, unspanned records in ASCII type D format are processed.

*DB Variable length, blocked, unspanned records in ASCII type D format are processed.

*VS Variable length, deblocked, spanned records in either EBCDIC or ASCII code are processed.

*VBS Variable length, blocked, spanned records in either EBCDIC or ASCII code are processed. The representation of spanned records on the tape is different for EBCDIC and ASCII files, but the system selects the correct format based on the file code.

*U Undefined format records in either EBCDIC or ASCII code are processed.
### Table 4. Figure: Required RCDLEN/BLKLEN/BUFOFSET Relation

<table>
<thead>
<tr>
<th>CODE</th>
<th>RCDLKFMT</th>
<th>BLKLEN</th>
</tr>
</thead>
<tbody>
<tr>
<td>== ===</td>
<td>=RCDLEN</td>
<td>=RCDLEN+BUFOFSET</td>
</tr>
<tr>
<td>*EBCDIC</td>
<td>+F +U</td>
<td>=RCDLEN</td>
</tr>
<tr>
<td>*ASCII</td>
<td>+F +U</td>
<td>=RCDLEN+BUFOFSET</td>
</tr>
<tr>
<td>*EBCDIC</td>
<td>+FB</td>
<td>=RCDLEN+n</td>
</tr>
<tr>
<td>*ASCII</td>
<td>+FB</td>
<td>=RCDLEN+n+BUFOFSET</td>
</tr>
<tr>
<td>(where n is the number of records in a maximum-length block)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>*EBCDIC</td>
<td>+V</td>
<td>=RCDLEN+4</td>
</tr>
<tr>
<td>*ASCII</td>
<td>+D</td>
<td>=RCDLEN+4+BUFOFSET</td>
</tr>
<tr>
<td>*EBCDIC</td>
<td>+VS +VBS</td>
<td>&gt;= 18</td>
</tr>
<tr>
<td>*ASCII</td>
<td>+BS +VBS</td>
<td>&gt;= 6+BUFOFSET (18 minimum)</td>
</tr>
</tbody>
</table>

**NOTE:** Block length (BLKLEN) is a function of record length (RCDLEN) and buffer offset (BUFOFSET).

---

## Extend file (EXTEND)

Specifies, for output operations to tape, whether new records are added to the end of a data file that is currently on the tape. If the data file is extended, it becomes the last file on the tape volume.

**Note:** This parameter is not valid for 1/4-inch cartridge tape devices.

### Single values

- **NO** Records are not added to the end of the specified data file.

### Element 1: Extend file

- **YES** New records are added to the end of the specified data file on tape when this device file is used.

### Element 2: Check file

- **NOCHECK** The file is extended without being checked to determine whether it is active.

- **CHECK** Before the file is extended, it is checked to determine whether it is active.

---

## Tape density (DENSITY)

Specifies the density of the data that is written on the tape volume when this device file is created. This parameter is used only for tape files being written to tape; it is ignored for tape files being read from the tape (in the case of files being read from tape, the density on the tape is used).

The density of a standard-label volume is specified on the INZTAP command, which initializes tapes as standard-label volumes by writing volume labels on them. If the density specified for this parameter is different than the density of a standard-labeled tape, the tape must be reinitialized to the specified density.
*DEVTYPE

The highest capacity density or format supported by the tape device will be used.

<table>
<thead>
<tr>
<th>Device</th>
<th>Highest capacity density or format</th>
</tr>
</thead>
<tbody>
<tr>
<td>3480</td>
<td>*FMT3480</td>
</tr>
<tr>
<td>3490E</td>
<td>*FMT3490E</td>
</tr>
<tr>
<td>3570-Bxx</td>
<td>*FMT3570</td>
</tr>
<tr>
<td>3570-Cxx</td>
<td>*FMT3570E</td>
</tr>
<tr>
<td>3580-001</td>
<td>*ULTRIUM1</td>
</tr>
<tr>
<td>3580-002</td>
<td>*ULTRIUM2</td>
</tr>
<tr>
<td>3580-003</td>
<td>*ULTRIUM3</td>
</tr>
<tr>
<td>3590-Bxx</td>
<td>*FMT3590</td>
</tr>
<tr>
<td>3590-Exx</td>
<td>*FMT3590E</td>
</tr>
<tr>
<td>3590-Hxx</td>
<td>*FMT3590H</td>
</tr>
<tr>
<td>3592-J1A</td>
<td>*FMT3592A1</td>
</tr>
<tr>
<td>4685-001</td>
<td>*VXA2</td>
</tr>
<tr>
<td>6335</td>
<td>*QIC3040</td>
</tr>
<tr>
<td>6343</td>
<td>*QIC1000</td>
</tr>
<tr>
<td>6344</td>
<td>*QIC2GB</td>
</tr>
<tr>
<td>6348</td>
<td>*QIC1000</td>
</tr>
<tr>
<td>6349</td>
<td>*QIC2GB</td>
</tr>
<tr>
<td>6368</td>
<td>*QIC1000</td>
</tr>
<tr>
<td>6369</td>
<td>*QIC2GB</td>
</tr>
<tr>
<td>6379</td>
<td>*QIC1000</td>
</tr>
<tr>
<td>6380</td>
<td>*QIC2GB</td>
</tr>
<tr>
<td>6381</td>
<td>*QIC2DC</td>
</tr>
<tr>
<td>6382</td>
<td>*QIC4DC</td>
</tr>
<tr>
<td>6383</td>
<td>*QIC5010</td>
</tr>
<tr>
<td>6384</td>
<td>*SLR60</td>
</tr>
<tr>
<td>6385</td>
<td>*QIC5010</td>
</tr>
</tbody>
</table>
The highest capacity density or format supported by the device for the mounted cartridge type will be used. If the device does not support special cartridge type information, *DEVTYPE is used.

**character-value**
Specify the density or format to use.

**1600** The data density on the tape volume is 1,600 bits per inch, which is used for 1/2 inch reel tapes.

**3200** The data density on the tape volume is 3,200 bits per inch, which is used for 1/2 inch reel tapes.

**6250** The data density on the tape volume is 6,250 bits per inch, which is used for 1/2 inch reel tapes.

**FMT3480**
The format of this tape is FMT3480. The data density on this tape volume is formatted to support a 3480 device. This density is used for 1/2 inch cartridge tapes.

**FMT3490E**
The format of this tape is FMT3490E. The data density on this tape volume is formatted to support a 3490E device. This density is used for 1/2 inch cartridge tapes.

**FMT3570**
The format of this tape is FMT3570. The data format is written on the tape volume with a 3570 device.

**FMT3570E**
The format of this tape is FMT3570E. The data format is written on the tape volume with a 3570E device.

**FMT3590**
The format of this tape is FMT3590. The data format is written on the tape volume with a 3590 device. This density is used for 1/2 inch cartridge tapes.
*FMT3590E
The format of this tape is FMT3590E. The data format is written on the tape volume with a 3590E device. This density is used for 1/2 inch cartridge tapes.

*FMT3590H
The format of this tape is FMT3590H. The data format is written on the tape volume with a 3590H device. This density is used for 1/2 inch cartridge tapes.

*FMT3592A1
The format of this tape is FMT3592A1. The data format is written on the tape volume with a 3592 device. This density is used for 1/2 inch cartridge tapes.

*QIC120
The format of this tape is QIC120, which is used for 1/4 inch cartridge tapes that can hold 120 megabytes of data.

*QIC525
The format of this tape is QIC525, which is used for 1/4 inch cartridge tapes that can hold 525 megabytes of data.

*QIC1000
The format of this tape is QIC1000, which is used for 1/4 inch cartridge tapes that can hold 1200 megabytes of data.

*QIC2GB
The format of this tape is QIC2GB. It is used by 1/4 inch tape devices which can store 2.5 gigabytes of data on a standard length QIC2GB cartridge.

*QIC2DC
The format of this tape is QIC2DC. It is used to write compacted data to a 1/4 inch cartridge that supports the QIC2GB format.

*QIC4GB
The format of this tape is QIC4GB. It is used by 1/4 inch tape devices which can store 4 gigabytes of data on a standard length QIC4GB cartridge.

*QIC4DC
The format of this tape is QIC4DC. It is used to write compacted data to a 1/4 inch cartridge that supports the QIC4GB format.

*QIC3040
The format of this tape is QIC3040, which is used for 1/4 inch minicartridge tapes that can hold 840 megabytes of data.

*QIC5010
The format of this tape is QIC5010, which is used for 1/4 inch cartridge tapes that can hold 13.5 gigabytes of data.

*MLR3
The format of this tape is MLR3. It is used by 1/4 inch tape devices which can store 25 gigabytes of data on a standard length MLR3 cartridge.

*SLR60
The format of this tape is SLR60. It is used by 1/4 inch tape devices which can typically store 60 gigabytes of compacted data on a standard length SLR60 cartridge.

*SLR100
The format of this tape is SLR100. It is used by 1/4 inch tape devices which can typically store 100 gigabytes of compacted data on a standard length SLR100 cartridge.

*FMT2GB
The format of this tape is FMT2GB, which is used for 8 millimeter cartridge tapes that can hold 2 gigabytes of data.
*FMT5GB
The format of this tape is FMT5GB, which is used for 8 millimeter cartridge tapes that can hold 5 gigabytes of data.

*FMT7GB
The format of this tape is FMT7GB, which is used for 8 millimeter cartridge tapes that can hold 7 gigabytes of data.

*FMT20GB
The format of this tape is FMT20GB. It is used by 8 millimeter tape devices that can store 20 gigabytes of data on a standard length cartridge.

*FMT60GB
The format of this tape is FMT60GB. It is used by 8 millimeter tape devices that can store 60 gigabytes of data on a standard length cartridge.

*ULTRIUM1
The format of this tape is ULTRIUM1. It is used by 1/2 inch cartridge tape devices that can store 100 gigabytes of data on a standard length cartridge.

*ULTRIUM2
The format of this tape is ULTRIUM2. It is used by 1/2 inch cartridge tape devices that can store 200 gigabytes of data on a standard length cartridge.

*ULTRIUM3
The format of this tape is ULTRIUM3. It is used by 1/2 inch cartridge tape devices that can store 400 gigabytes of data on a standard length cartridge.

*VRT32K
The format of the volume is VRT32K. It is used to write data to a virtual volume using a maximum data block size of 32KB. Volumes written using this format can be duplicated to all supported tape devices.

*VRT64K
The format of the volume is VRT64K. It is used to write data to a virtual volume using a maximum data block size of 64KB. Volumes written using this format can only be duplicated to tape devices that support a maximum block size of 64KB or greater.

*VRT240K
The format of the volume is VRT240K. It is used to write data to a virtual volume using a maximum data block size of 240KB. Volumes written using this format can only be duplicated to tape devices that support a maximum block size of 240KB or greater.

*VRT256K
The format of the volume is VRT256K. It is used to write data to a virtual volume using a maximum data block size of 256KB. Volumes written using this format can only be duplicated to tape devices that support a maximum block size of 256KB or greater.

*VXA1
The format of this tape is VXA1. It is used by VXA cartridge tape devices that can store 33 gigabytes of data on a standard length cartridge.

*VXA2
The format of this tape is VXA2. It is used by VXA cartridge tape devices that can store 80 gigabytes of data on a standard length cartridge.

Note: Self-configured tape devices may define additional valid values for the density parameter. Use iSeries Navigator (Configuration and Service) (Hardware) (Tape Devices) (Tape Libraries) (Tape Resources) (Properties) or (Configuration and Service)(Hardware) (Tape Devices) (Stand-Alone Devices) (Properties) to find additional valid density values for a specific device, or use the F4=Prompt key on the "Tape density" field of the CL command to see a list of all valid density values for the attached tape devices.
**Data compaction (COMPACT)**

 Specifies whether device data compaction is performed. If the tape devices being used do not support data compaction, this parameter will be ignored when the file is opened.

 - **DEV**  
  Device data compaction is performed if the devices being used support data compaction.

 - **NO**  
  Device data compaction is not performed.

**Code (CODE)**

 Specifies the type of character code used when tape data is read or written by a job that uses this tape device file.

 - **EBCDIC**  
  The EBCDIC character code is used with this tape device file.

 - **ASCII**  
  The ASCII character code is used.

**Creation date (CRTDATE)**

 Specifies the date when the data file was created on (written to) tape. The data file creation date is stored in file labels on the tape. If a creation date is specified for any type of label processing other than *SL, it is ignored.

 - **NONE**  
  The creation date is not specified.

 - **date**  
  Specify the creation date of the data file used by this tape device file.

**File expiration date (EXPDATE)**

 Specifies, for tape output data files only, the expiration date of the data file used by this device file. The data file expiration date is stored in file labels on the tape. If an expiration date is specified for any type of label processing other than *SL, it is ignored. The data file is protected and cannot be written over until the specified expiration date.

 - **NONE**  
  No expiration date is specified for the data file. The data file is not protected.

 - **PERM**  
  The data file is protected permanently. The date written on the tape is 999999.

 - **date**  
  Specify the date on which and after which the data file is no longer protected.
End of tape option (ENDOPT)

Specifies the positioning operation performed automatically on the tape volume when the tape device file is closed. In the case of a multiple-volume data file, this parameter applies only to the last reel.

*RE WIND
The tape is automatically rewound, but not unloaded, after the operation has ended.

*UNLOAD
The tape is automatically rewound and unloaded after the operation ends.

*LEAVE
The tape does not rewind or unload after the operation ends. It remains at the current position on the tape drive.

Note: Even if *LEAVE is specified for the ENDOPT parameter, the next tape file opened to this reel is positioned at the start or end of a data file when it is opened.

User label program (USRLBLPGM)

Specifies the user program that processes user-defined tape labels. On an output file, the user label program passes the user labels that are written to tape. On an input file, the user labels are passed to the user label program.

Single values

*NONE
There is no user label program for this device file.

Qualifier 1: User label program

name Specify the name of the user program that processes the user tape labels.

Qualifier 2: Library

*LIBL All libraries in the library list for the current thread are searched until the first match is found.

*CURLIB
The current library for the thread is searched. If no library is specified as the current library for the thread, the QGPL library is searched.

name Specify the name of the library to be searched.

Maximum file wait time (WAITFILE)

Specifies the number of seconds that the program waits for the file resources to be allocated when the file is opened, or the device or session resources to be allocated when an acquire operation is performed to the file. If the file resources cannot be allocated in the specified wait time, an error message is sent to the program.

*IMMED
The program does not wait. Immediate allocation of file resources is required.

*CLS The job default wait time is used as the wait time for the file resources to be allocated.
Specify the number of seconds to wait for file resources to be allocated.

**Share open data path (SHARE)**

Specifies whether the open data path (ODP) is shared with other programs in the same routing step. When an ODP is shared, the programs accessing the file share facilities such as the file status and the buffer.

*NO  The ODP is not shared with other programs in the routing step. A new ODP for the file is created and used every time a program opens the file.

*YES The same ODP is shared with each program in the job that also specifies *YES when it opens the file.

**Authority (AUT)**

Specifies the authority you are giving to users who do not have specific authority for the object, who are not on an authorization list, and whose group profile or supplemental group profiles do not have specific authority for the object.

*LIBCRTAUT  The system determines the authority for the object by using the value specified for the Create authority (CRTAUT) parameter on the Create Library command (CRTLIB) for the library containing the object to be created. If the value specified for the CRTAUT parameter is changed, the new value will not affect any existing objects.

*CHANGE  The user can perform all operations on the object except those limited to the owner or controlled by object existence (*OBJEXIST) and object management (*OBJMGT) authorities. The user can change and perform basic functions on the object. *CHANGE authority provides object operational (*OBJOPR) authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove users.

*ALL  The user can perform all operations except those limited to the owner or controlled by authorization list management (*AUTLMGT) authority. The user can control the object’s existence, specify the security for the object, change the object, and perform basic functions on the object. The user also can change ownership of the object.

*USE  The user can perform basic operations on the object, such as running a program or reading a file. The user cannot change the object. Use (*USE) authority provides object operational (*OBJOPR), read (*READ), and execute (*EXECUTE) authorities.

*EXCLUDE  The user cannot access the object.

name  Specify the name of an authorization list to be used for authority to the object. Users included in the authorization list are granted authority to the object as specified in the list. The authorization list must exist when the object is created.
Replace file (REPLACE)

Specifies whether an existing file, other than a save or database file, is replaced.

**YES**  An file is replaced if the creation of the new tape device file with the same name and library is successful.

**NO**  The creation of a new tape device file is not allowed if there is an existing file with the same name and library.

Examples

**Example 1: Creating a Description of a Tape Device File**

```bash
CRTTAPF  FILE(BACKHST)  DEV(QTAPE1 QTAPE2 QTAPE3)
         REELS(*BLP 10)  RCDLEN(256)  BLKLEN(1024)
         RCDBLKFM(*FB)  EXTEND(*YES)
         ENDOPT(*UNLOAD)  WAITFILE(60)
```

This command creates a description of the tape device file named BACKHST in the current library, to be used with the tape devices QTAPE1, QTAPE2, and QTAPE3. All volumes processed on these devices with this device file must have standard-labels. Each block of data (EBCDIC character code) on the tape volumes contains four records of 256 bytes each. When records are written to the tape, they are added to the end of the data file. No creation or expiration date is specified for this tape, and both unloading and rewinding operations will occur when the device file is closed at the last tape volume processed. The program using this tape device file waits 60 seconds for file resources to be allocated when this file is opened, and this device file is used only when the current program is running.

**Example 2: Creating a Tape File Containing DBCS Data**

```bash
CRTTAPF  FILE(IGCLIB/IGCTAP)  LABEL(GENINF)  IGCDTA(*YES)
```

This command creates the tape file IGCTAP that is stored in library IGCLIB, which is labeled GENINF, and contains DBCS data.

Error messages

**ESCAPE Messages**

CPF7302  
  File &1 not created in library &2.
Create Table (CRTTBL)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Create Table (CRTTBL) command creates a named table. The table can be used for the translation of data that is moved between the system and a device. The table can also be used to specify an alternate collating sequence or to specify field translation functions.

You can create a table by prompting for the source information, or by specifying a source member.

To create a table using prompting support, specify *PROMPT on the SRCFILE parameter. A display is shown allowing you to view and change the table values and a new table is created based on the values chosen on the display.

To create a *C VT table using a source member, you must specify a source member that contains 8 records. Each record in this member must contain 64 hexadecimal characters (characters after position 64 are not used). This input (512 hexadecimal characters) is converted and stored internally as 256 bytes by the CRTTBL command.

The value you specify for a position within the source member is the same value that is returned by QDCXLATE (or other system program) whenever that position is encountered.

For example, if you specify "C0C1C2C3C4C5C6..." as the first part of record 1 in the source, then a hexadecimal "C0" is returned when given a hexadecimal "00", a hexadecimal "C1" is returned when given a hexadecimal "01" and so on.

To create a *UCSSRTSEQ table using a source member, you must specify a source member that contains the following layout of information.
1. column 1-4 = Hex code point to be sorted
2. column 6-10 = Weight of this code point as a decimal number
3. column 11-80 = not used, can contain such things as comments.

### Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>TBL</strong></td>
<td>Table</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Table</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *CURLIB</td>
<td></td>
</tr>
<tr>
<td><strong>SRCFILE</strong></td>
<td>Source file</td>
<td>Single values: *PROMPT Other values: Qualified object name</td>
<td>Optional, Positional 2</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Source file</td>
<td>Name, QTBLSRC</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
</tr>
<tr>
<td><strong>SRCMBR</strong></td>
<td>Source member</td>
<td>Name, *TBL</td>
<td>Optional, Positional 3</td>
</tr>
<tr>
<td><strong>TBLTYPE</strong></td>
<td>Table type</td>
<td>*C VT, *SRTSEQ, *UCSSRTSEQ</td>
<td>Optional</td>
</tr>
</tbody>
</table>
Table (TBL)

Specifies the library name and table name of the table being created.

This is a required parameter.

The possible library values are:

* CURLIB
  The library for the job is used to locate the table. If no library is specified as the library for the job, QGPL is used.

name  Specify the name of the library to be searched.

The possible values are:

  table-name
  Specify the name of the table being created.

Source file (SRCFILE)

Specifies the qualified name of the source file containing the description of the table being created or that prompting support is to be used.

The possible values are:

* PROMPT
  Allows the user to access and use source information.

QTBLSRC
  The system source file named QTBLSRC contains the source records that are used with this command to create the table.

source-file-name
  Specify the name (library-name/source-file-name) of the source file that contains the source records that are used with this command to create the table.
The possible library values are:

**LIBL** All libraries in the library list for the current thread are searched until the first match is found.

**CURLIB** The current library for the job is used to locate the source file. If no current library is specified as the current library for the job, QGPL is used.

**name** Specify the name of the library to be searched.

---

**Source member (SRCMBR)**

Specifies the name of the source file member containing the description of the table being created.

The possible values are:

**TBL** The source file member name is the same as the name of the table.

**source-file-member-name** Specify the name of the member in the source file specified on the Source file prompt (SRCFILE parameter) to create the table.

**generic*-table-name** Specify the generic name of the table. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. If a generic name is specified, then all tables with names that begin with the generic name, and for which the user has authority, are shown. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete table name.

---

**Table type (TBLTYPE)**

Specifies the type of table object to be created.

The possible values are:

**CVT** A conversion type table is created.

**SRTSEQ** A sort sequence type table is created.

**UCSSRTSEQ** An ISO-10646 UCS-2 sort sequence type table is created.

---

**Basing table (BASETBL)**

Specifies the base table to be used for prompting support when creating a conversion table.

The possible values are:

**HEX** A one to one mapping table is used.

The name of the table can be qualified by one of the following library values:

**LIBL** All libraries in the library list for the current thread are searched until the first match is found.
*CURLIB  
The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name  
Specify the name of the library to be searched.

table-name  
Specify a table object to be used as a base.

---

**Basing sort sequence (BASESRTSEQ)**

Specifies the base table to be used for prompting support when creating a sort sequence table.

The possible values are:

- ***JOB**  The sort sequence must be resolved when the object is created.
- **LANGIDSHR**  A shared weight sort table is used.
- **LANGIDUNQ**  A unique weight sort table is used.
- ***HEX**  A sort sequence table is not used. The hexadecimal values of the characters are used to determine the sort sequence. The CCSID for hexadecimal data is 65535.

The name of the table can be qualified by one of the following library values:

- ***LIBL**  All libraries in the library list for the current thread are searched until the first match is found.
- ***CURLIB**  The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

library-name  
Specify the name of the library to be searched.

table-name  
Specify the name of a table object to be used as a base.

---

**Basing language ID (BASELANGID)**

Specifies the base language used for prompting support when creating a sort sequence table.

The possible values are:

- ***JOB**  The sort sequence must be resolved when the object is created.
- **language-ID**  Specify a valid language ID to be used for the sort sequence table being created.
**Coded character set ID (CCSID)**

Specifies the coded character set ID (CCSID) in which to store the sort sequence table information.

The possible values are:

*JOB The sort sequence must be resolved when the object is created.

coded-character-set-ID Specify the CCSID to be used for the sort sequence table information.

---

**Text ’description’ (TEXT)**

Specifies the text that briefly describes the object.

The possible values are:

*BLANK No text is specified.

caller-value Specify no more than 50 characters of text, enclosed in apostrophes.

---

**Authority (AUT)**

Specifies the authority you are giving to users who do not have specific authority for the object, who are not on an authorization list, and whose group profile or supplemental group profiles do not have specific authority for the object.

**LIBCRTAUT**

The authority for the object is the same as the value specified on the Create authority prompt (CRTAUT parameter) of the library in which the object is being created. If the value specified on the Create authority prompt (CRTAUT parameter) is changed, the new value will not affect any existing objects.

**CHANGE**

The user can perform all operations on the object except those limited to the owner or controlled by object existence (*OBJEXIST) and object management (*OBJMGT) authorities. The user can change and perform basic functions on the object. *CHANGE authority provides object operational (*OBJOPR) authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove users.

**ALL**

The user can perform all operations except those limited to the owner or controlled by authorization list management (*AUTLMGT) authority. The user can control the object’s existence, specify the security for the object, change the object, and perform basic functions on the object. The user also can change ownership of the object.

**USE**

The user can perform basic operations on the object, such as running a program or reading a file. The user cannot change the object. Use *USE authority provides object operational (*OBJOPR), read (*READ), and execute (*EXECUTE) authorities.
The user cannot access the object.

authorization-list-name

Specify the name of an authorization list to be used for authority to the object. Users included in the authorization list are granted authority to the object as specified in the list. The authorization list must exist when the object is created.

Examples

CRTLBL TBL(SCRAMTBL) SRCFILE(USERTABLES) SRCMBR(SCRAMBLE)
TEXT('Translate table for scrambling text characters')

This command creates a table named SCRAMTBL and (by default) stores it in the current library. The source file named USERTABLES contains the source records used when the table is created; the name of the source file member is SCRAMBLE. The TEXT parameter describes this table as being used as a translate table for scrambling text characters.

Error messages

*ESCAPE Messages

CPF2207
Not authorized to use object &1 in library &3 type *&2.

CPF2614
Table &1 in &2 already exists.

CPF2623
Library &1 not found.

CPF2678
Data in source file &1 in &2 not valid.

CPF3BF7
Data in source file at line &1 is not correct.

CPF3FC9
Value &1 for CCSID not valid.

CPF9820
Not authorized to use library &1.
Create Time Zone Description (CRTTIMZON)

Where allowed to run: All environments (*ALL)
Threadsafe: Yes

The Create Time Zone Description (CRTTIMZON) command creates a time zone description object that defines the properties of a time zone. The time zone description object is created in the QSYS library. These properties are used to convert time values between Coordinated Universal Time (UTC) form and local forms. These properties are also used to express time values in local forms.

Restrictions:
• You must have read (*READ) and add (*ADD) authorities to the QSYS library.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>TIMZON</td>
<td>Time zone description</td>
<td>Name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td>OFFSET</td>
<td>Offset</td>
<td>-779-779</td>
<td>Required, Positional 2</td>
</tr>
<tr>
<td>STDNAME</td>
<td>Standard Time</td>
<td>Single values: *GEN, *MSG</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Element 1: Abbreviated name</td>
<td>Character value</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Full name</td>
<td>Character value</td>
<td></td>
</tr>
<tr>
<td>DSTNAME</td>
<td>Daylight Saving Time (DST)</td>
<td>Single values: *NONE, *GEN, *MSG</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Element 1: Abbreviated name</td>
<td>Character value</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Full name</td>
<td>Character value</td>
<td></td>
</tr>
<tr>
<td>STDMSG</td>
<td>Standard Time message</td>
<td>Name</td>
<td>Optional</td>
</tr>
<tr>
<td>DSTMSG</td>
<td>Daylight Saving Time message</td>
<td>Name</td>
<td>Optional</td>
</tr>
<tr>
<td>MSGF</td>
<td>Message file</td>
<td>Qualified object name</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Message file</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL</td>
<td></td>
</tr>
<tr>
<td>DSTSTR</td>
<td>Daylight Saving Time start</td>
<td>Element list</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Element 3: Relative day of month</td>
<td>LAST, 1, 2, 3, 4</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 4: Time</td>
<td>Time</td>
<td></td>
</tr>
</tbody>
</table>
**Keyword** | **Description** | **Choices** | **Notes**
---|---|---|---
DSTEND | Daylight Saving Time end | *Element list* | Optional
Element 3: Relative day of month | *LAST, 1, 2, 3, 4 |
Element 4: Time | *Time |

**TEXT** | Text ‘description’ | *Character value, *BLANK | Optional

**AUT** | Authority | *Name, *USE, *LIBCRTAUT, *CHANGE, *ALL, *EXCLUDE | Optional

---

**Time zone description (TIMZON)**

Specifies the time zone description to be created. The time zone description is created in QSYS.

This is a required parameter.

*name*  Specify the name of the time zone description.

---

**Offset (OFFSET)**

Specifies the time difference, in minutes, between this time zone and Coordinated Universal Time (UTC). This value is subtracted from local time to obtain UTC time. A negative difference indicates that the time zone is west of UTC and a positive difference indicates that the time zone is east of UTC.

This is a required parameter.

*779 to 779*  
Specify the time difference, in minutes. Valid values range from -779 minutes to 779 minutes.

---

**Standard Time (STDNAME)**

Specifies the abbreviated and full names of the time zone when Daylight Saving Time is not being observed.

**Single values**

*GEN*  
The system will generate the abbreviated and full names. The format of the abbreviated name will be the letters ‘UTC’ followed by the offset followed by the letter ‘S’. The offset will appear as a formatted hour and minute value. The full name for the time zone description will be the same as the abbreviated name. For example, a time zone that has an offset of -360 minutes would have an abbreviated and a full name of ‘UTC-06:00S’.

*MSG*  
The abbreviated and full names will be retrieved from the second-level message text of the message specified for the Standard Time message (STDMSG) parameter. When this value is specified, values must also be specified for the STDMSG parameter and the Message file (MSGF) parameter.
Element 1: Abbreviated name

count-character-value

Specify the abbreviated or short name for this time zone. The abbreviated name has a maximum length of 10 characters.

Element 2: Full name

count-character-value

Specify the full or long name for this time zone. The full name has a maximum length of 50 characters.

Daylight Saving Time (DST) (DSTNAME)

Specifies the abbreviated and full names of the time zone when Daylight Saving Time is being observed.

Single values

*NONE

This time zone does not observe Daylight Saving Time.

*GEN

The system will generate the abbreviated and full names. The format of the abbreviated name will be the letters 'UTC' followed by the offset followed by the letter 'D'. The offset will appear as a formatted hour and minute value. The full name for the time zone description will be the same as the abbreviated name. For example, a time zone that has an offset of -360 minutes would have an abbreviated and a full name of 'UTC-06:00D'.

*MSG

The abbreviated and full names will be retrieved from the second-level message text of the message specified for the Daylight Saving Time message (DSTMSG) parameter. When this value is specified, values must also be specified for the DSTMSG parameter and the Message file (MSGF) parameter.

Element 1: Abbreviated name

count-character-value

Specify the abbreviated or short name for this time zone. The abbreviated name has a maximum length of 10 characters.

Element 2: Full name

count-character-value

Specify the full or long name for this time zone. The full name has a maximum length of 50 characters.

Standard Time message (STDMSG)

Specifies the predefined message that contains the abbreviated and full names of the time zone that are used when Daylight Saving Time is not being observed. The first 10 characters of the message contain the abbreviated name and the next 50 characters contain the full name. This parameter must be specified when *MSG is specified for the Standard Time (STDNAME) parameter.

name

Specify the message identifier.
Daylight Saving Time message (DSTMSG)

Specifies the predefined message that contains the abbreviated and full names of the time zone that are used when Daylight Saving Time is being observed. The first 10 characters of the message contain the abbreviated name and the next 50 characters contain the full name. This parameter must be specified when *MSG is specified for the Daylight Saving Time (DST) (DSTNAME) parameter.

name Specify the message identifier.

Message file (MSGF)

Specifies the message file from which the Standard Time message and the Daylight Saving Time message are to be retrieved. The specified message file name and library name are stored in the time zone description. When a message is used to specify the abbreviated and full names, the message is retrieved each time the abbreviated or full names are retrieved. If the message cannot be retrieved from the message file, the names will be returned as *N. This parameter must be specified when *MSG is specified for the Standard Time (STDNAME) parameter or the Daylight Saving Time (DST) (DSTNAME) parameter.

Qualifier 1: Message file

name Specify the name of the message file.

Qualifier 2: Library

*LIBL All libraries in the thread’s library list are searched for the message file when the message is retrieved. The value *LIBL is saved in the time zone description and is not resolved to a library name by this command.

name Specify the library where the message file is located.

Daylight Saving Time start (DSTSTR)

Specifies when Daylight Saving Time (DST) starts. This parameter contains four elements: the month in which DST starts, the day on which DST starts, the relative day of the month on which DST starts and the time at which DST starts. If this parameter is specified, all four elements must be specified. This parameter must be specified when a value other than *NONE is specified for the Daylight Saving Time (DST) (DSTNAME) parameter. The values specified for this parameter cannot be identical to the values specified for the Daylight Saving Time end (DSTEND) parameter.

Element 1: Month

*JAN Daylight Saving Time starts in January.
*FEB Daylight Saving Time starts in February.
*MAR Daylight Saving Time starts in March.
*APR Daylight Saving Time starts in April.
*MAY Daylight Saving Time starts in May.
*JUN Daylight Saving Time starts in June.
*JUL Daylight Saving Time starts in July.
*AUG Daylight Saving Time starts in August.
*SEP  Daylight Saving Time starts in September.
*OCT  Daylight Saving Time starts in October.
*NOV  Daylight Saving Time starts in November.
*DEC  Daylight Saving Time starts in December.

Element 2: Day

*MON  Daylight Saving Time starts on a Monday.
*TUE  Daylight Saving Time starts on a Tuesday.
*WED  Daylight Saving Time starts on a Wednesday.
*THU  Daylight Saving Time starts on a Thursday.
*FRI  Daylight Saving Time starts on a Friday.
*SAT  Daylight Saving Time starts on a Saturday.
*SUN  Daylight Saving Time starts on a Sunday.

Element 3: Relative day of month

*LAST  Daylight Saving Time starts on the last occurrence of the specified day of the week.
1  Daylight Saving Time starts on the first occurrence of the specified day of the week.
2  Daylight Saving Time starts on the second occurrence of the specified day of the week.
3  Daylight Saving Time starts on the third occurrence of the specified day of the week.
4  Daylight Saving Time starts on the fourth occurrence of the specified day of the week.

Element 4: Time

time  Specify the time of day at which Daylight Saving Time starts. The time is specified in 24-hour format and can be specified with or without a time separator.

- Without a time separator, specify a string of 4 or 6 digits (hhmm or hhmmss) where hh = hours, mm = minutes, and ss = seconds. Valid values for hh range from 00 to 23. Valid values for mm and ss range from 00 to 59.
- With a time separator, specify a string of 5 or 8 characters where the time separator specified for your job is used to separate the hours, minutes, and seconds. If this command is entered from the command line, the string must be enclosed in apostrophes. If a time separator other than the separator specified for your job is used, this command will fail.

Daylight Saving Time end (DSTEND)

Specifies when Daylight Saving Time (DST) ends. This parameter contains four elements: the month in which DST ends, the day on which DST ends, the relative day of the month on which DST ends and the time at which DST ends. If this parameter is specified, all four elements must be specified. This parameter must be specified when a value other than *NONE is specified for the Daylight Saving Time (DST) (DSTNAME) parameter. The values specified for this parameter cannot be identical to the values specified for the Daylight Saving Time start (DSTSTR) parameter.

Element 1: Month
Daylight Saving Time ends in January.

Daylight Saving Time ends in February.

Daylight Saving Time ends in March.

Daylight Saving Time ends in April.

Daylight Saving Time ends in May.

Daylight Saving Time ends in June.

Daylight Saving Time ends in July.

Daylight Saving Time ends in August.

Daylight Saving Time ends in September.

Daylight Saving Time ends in October.

Daylight Saving Time ends in November.

Daylight Saving Time ends in December.

**Element 2: Day**

Daylight Saving Time ends on a Monday.

Daylight Saving Time ends on a Tuesday.

Daylight Saving Time ends on a Wednesday.

Daylight Saving Time ends on a Thursday.

Daylight Saving Time ends on a Friday.

Daylight Saving Time ends on a Saturday.

Daylight Saving Time ends on a Sunday.

**Element 3: Relative day of month**

Daylight Saving Time ends on the last occurrence of the specified day of the week.

Daylight Saving Time ends on the first occurrence of the specified day of the week.

Daylight Saving Time ends on the second occurrence of the specified day of the week.

Daylight Saving Time ends on the third occurrence of the specified day of the week.

Daylight Saving Time ends on the fourth occurrence of the specified day of the week.

**Element 4: Time**

Specify the time of day at which Daylight Saving Time ends. The time is specified in 24-hour format and can be specified with or without a time separator.

Without a time separator, specify a string of 4 or 6 digits (hhmm or hhmmss) where hh = hours, mm = minutes, and ss = seconds. Valid values for hh range from 00 to 23. Valid values for mm and ss range from 00 to 59.

With a time separator, specify a string of 5 or 8 characters where the time separator specified for your job is used to separate the hours, minutes, and seconds. If this command is entered from the command line, the string must be enclosed in apostrophes. If a time separator other than the separator specified for your job is used, this command will fail.
Text 'description' (TEXT)

Specifies the text that briefly describes the object.

*BLANK

No text is specified.

character-value

Specify no more than 50 characters of text, enclosed in apostrophes.

Authority (AUT)

Specifies the authority you are giving to users who do not have specific authority to the object, who are not on an authorization list, and whose group profile has no specific authority to the object.

*USE

The user can perform basic operations on the time zone description, such as retrieving its contents. The user cannot change the time zone description. *USE authority provides object operational authority, read authority, and execute authority.

*LIBCRTAUT

The authority for the object is taken from the value specified for the Create authority (CRTAUT) parameter of the library in which the object is being created. If the value specified for the CRTAUT parameter is changed, the new value will not affect any existing objects.

*CHANGE

Change authority allows the user to change and perform basic functions on the object. The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. Change authority provides object operational authority and all data authority.

*ALL

The user performs all operations on the object except those limited to the owner.

*EXCLUDE

The user cannot access the object.

name

Specify the name of an authorization list to be used for authority to the object. Users included in the authorization list are granted authority to the object as specified in the list. The authorization list must exist when the object is created.

Examples

Example 1: Creating a Time Zone Description

CRTTIMZON TIMZON(CENTRAL) OFFSET(-360)
STDNAME(CTZ 'Central Time Zone')
TEXT('Central Time Zone with Standard Time Only')

This command creates the CENTRAL time zone description. The offset for the time zone is negative six hours (-360 minutes). The Standard Time abbreviated name of the time zone is 'CTZ' and the Standard Time full name of the time zone is 'Central Time Zone'. The text description associated with the time zone description object is 'Central Time Zone with Standard Time Only'. This time zone description does not observe Daylight Saving Time.

Example 2: Creating a Time Zone Description that Supports Daylight Saving Time
This command creates the CENTRALDST time zone description. The offset for the time zone is negative six hours (-360 minutes). The Standard Time abbreviated name of the time zone is 'CTZ' and the Standard Time full name of the time zone is 'Central Time Zone'. The Daylight Saving Time abbreviated name of the time zone is 'CDTZ' and the Daylight Saving Time full name is 'Central Daylight Time Zone'. The text description associated with the time zone description object is 'Central Time Zone with Daylight Saving Time'. This time zone description does observe Daylight Saving Time. Daylight Saving Time starts at 2:00 am on the first Sunday in April and ends at 2:00 am on the last Sunday in October.

**Error messages**

*ESCAPE Messages*

CPF09A1

Time zone description &1 not created.
Create User-Defined FS (CRTUDFS)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Create User-Defined File System (CRTUDFS) command creates a file system that can be made visible to the rest of the integrated file system name space through the Add Mounted File System (ADDMFS) or MOUNT command.

A UDFS is represented by the object type *BLKSF, or block special file.

Restrictions:
1. The user must have input/output (I/O) system configuration (*IOSYSCFG) special authority to use this command.
2. The audit (*AUDIT) special authority is required when specifying a value other than *SYSVAL on the Auditing value for objects (CRTOBJAUD) parameter.
3. The user must have all object (*ALLOBJ) and security administrator (*SECADM) special authorities to specify a value for the Scanning option for objects (CRTOBJSCAN) parameter other than *PARENT.
4. A maximum of approximately 4,000 user-defined file systems can be created on an independent auxiliary storage pool (ASP).

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>UDFS</td>
<td>User-defined file system</td>
<td>Path name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td>CRTOBJSCAN</td>
<td>Scanning option for objects</td>
<td>*PARENT, *YES, *NO, *CHGONLY</td>
<td>Optional</td>
</tr>
<tr>
<td>RSTDNRNMLT</td>
<td>Restricted rename and unlink</td>
<td>*NO, *YES</td>
<td>Optional</td>
</tr>
<tr>
<td>CASE</td>
<td>Case sensitivity</td>
<td>*MIXED, *MONO</td>
<td>Optional</td>
</tr>
<tr>
<td>DFTFILEFMT</td>
<td>Default file format</td>
<td>*TYPE1, *TYPE2</td>
<td>Optional</td>
</tr>
<tr>
<td>TEXT</td>
<td>Text ‘description’</td>
<td>Character value, *BLANK</td>
<td>Optional</td>
</tr>
</tbody>
</table>

User-defined file system (UDFS)

Specifies the path name of the file system to be created. It must be in one of the following two forms:
• `/dev/qaspXX/udfsname.udfs`, where `XX` is one of the valid system or basic user auxiliary storage pool (ASP) numbers on the system, and `udfsname` is the name of the user-defined file system. All other parts of the name must appear as in the example above.

• `/dev/aspname/udfsname.udfs`, where `aspname` is one of the valid independent ASP names on the system, and `udfsname` is the name of the user-defined file system. All other parts of the name must appear as in the example above.

The name part of the path must be unique within the specified `qaspXX` or `aspname` directory.

This is a required parameter.

**Public authority for data (DTAAUT)**

Specifies the public data authority given to the user for the new user-defined file system (UDFS), or specifies that all authorities are inherited from the directory it is to be created in.

*INDIR*

The authority for the UDFS to be created is determined by the directory it is to be created in. This means the new UDFS will inherit its primary group, authorization list, and its public, private and primary group authorities from the `/dev/qaspXX` or `/dev/aspname` directory. If the value *INDIR* is specified for either the **Public authority for object (OBJAUT)** parameter or the DTAAUT parameter, then *INDIR* must be specified for both parameters.

*RWX* The user can change the object and perform basic functions on the object except those limited to the owner or controlled by object existence (*OBJEXIST), object management (*OBJMGT), object alter (*OBJALTER) and object reference (*OBJREF) authority. Read, write, execute (*RWX) provides object operational (*OBJOPR) and all data authorities.

*RW* The user can view and change the contents of an object. Read, write (*RW) authority provides *OBJOPR and data read (*READ), add (*ADD), update (*UPD) and delete (*DLT) authorities.

*RX* The user can perform basic operations on the object, such as run a program or display the contents of a file. The user is prevented from changing the object. Read, execute (*RX) authority provides *OBJOPR and data *READ and *EXECUTE authorities.

*WX* The user can change the contents of an object and run a program or search a library or directory. Write, execute (*WX) authority provides *OBJOPR and data *READ, *UPD, *DLT, and *EXECUTE authorities.

*R* The user can view the contents of an object. Read (*R) authority provides *OBJOPR and data *READ authorities.

*W* The user can change the contents of an object. Write (*W) authority provides *OBJOPR and data *READ, *UPD, and *DLT authorities.

*X* The user can run a program or search a library or directory. Execute (*X) authority provides *OBJOPR and data *EXECUTE authorities.

*EXCLUDE* The user cannot access the object. The OBJAUT value must be *NONE, if this special value is used.

*NONE* The user is given no data authorities to the object. This value cannot be used with OBJAUT value of *NONE.

*authorization-list-name* The format of the authorization list name remains the current ten-character format. The OBJAUT value must be *NONE, if this special value is used.
Public authority for object (OBJAUT)

Specifies the public object authority given to users for the user-defined file system, or specifies that all authorities are inherited from the directory it is to be created in.

*INDIR
The object authority for the UDFS to be created is determined by the directory it is to be created in. This means the new UDFS will inherit its primary group, authorization list, and its public, private and primary group authorities from the /dev/qaspXX or /dev/aspname directory. If the value *INDIR is specified for either the OBJAUT parameter or the Public authority for data (DTAAUT) parameter, then *INDIR must be specified for both parameters.

*NONE
None of the other object authorities (*OBJEXIST, *OBJMGT, *OBJALTER or *OBJREF) are given to the users. If *EXCLUDE or an authorization list is specified for the DTAAUT parameter, *NONE must be specified. This value cannot be used with the DTAAUT value of *NONE.

*ALL
All of the other object authorities (*OBJEXIST, *OBJMGT, *OBJALTER or *OBJREF) are given to the users.

The user can specify up to four of the following values:

*OBJEXIST
The user is given object existence (*OBJEXIST) authority to the object. The user can delete the object, free storage of the object, perform save and restore operations for the object, and transfer ownership of the object.

*OBJMGT
The user is given object management (*OBJMGT) authority to the object. With this authority the user can specify security for the object, move or rename the object and add members to database files.

*OBJALTER
The user is given object alter (*OBJALTER) authority to the object. The user is able to alter the attributes of the objects. On a database file, the user can add and remove triggers, add and remove referential and unique constraints, and change the attributes of the database file. With this authority on an SQL package, the user can change the attributes of the SQL package. Currently, this authority is used only for database files and SQL packages.

*OBJREF
The user is given object reference (*OBJREF) authority to objects. Used only for database files, the user can reference an object from another object such that operations on that object may be restricted by the other object. On a physical file, the user can add a referential constraint in which the physical file is the parent.

Auditing value for objects (CRTOBJJAUD)

Specifies the auditing value of root directory objects created in this user-defined file system.

*SYSVAL
The object auditing value for the objects in the UDFS is determined by the Create object auditing (QCRTOBJAUD) system value.

*NONE
Using or changing this object does not cause an audit entry to be sent to the security journal.
**USRPRF**

The user profile of the user accessing this object is used to determine if an audit record is sent for this access. The OBJAUD parameter of the Change User Auditing (CHGUSRAUD) command is used to turn on auditing for a specific user.

**CHANGE**

All change accesses to this object by all users are logged.

**ALL** All change or read accesses to this object by all users are logged.

---

**Scanning option for objects (CRTOBJSCAN)**

Specifies whether the root directory objects created in the user-defined file system will be scanned when exit programs are registered with any of the integrated file system scan-related exit points.

The integrated file system scan-related exit points are:

- QIBM_QP0L_SCAN_OPEN - Integrated File System Scan on Open Exit Program
- QIBM_QP0L_SCAN_CLOSE - Integrated File System Scan on Close Exit Program

For details on these exit points, see the System API Reference information in the iSeries Information Center at http://www.ibm.com/eserver/iseries/infocenter.

Even though this attribute can be set for user-defined file systems, only objects which are in *TYPE2 directories in that user-defined file system will actually be scanned, no matter what value is set for this attribute.

**PARENT**

The create object scanning attribute value for this user-defined file system is copied from the create object scanning attribute value of the parent directory.

**YES** After an object is created in the user-defined file system, the object will be scanned according to the rules described in the scan-related exit programs if the object has been modified or if the scanning software has been updated since the last time the object was scanned.

**NO** After an object is created in the user-defined file system, the object will not be scanned by the scan-related exit programs.

**Note:** If the Scan file systems control (QSCANFSCTL) value *NOPOSTRST is not specified when an object with this attribute is restored, the object will be scanned at least once after the restore.

**CHGONLY**

After an object is created in the user-defined file system, the object will be scanned according to the rules described in the scan-related exit programs only if the object has been modified since the last time the object was scanned. It will not be scanned if the scanning software has been updated. This attribute only takes effect if the Scan file systems control (QSCANFSCTL) system value has *USEOCOATR specified. Otherwise, it will be treated as if the attribute is *YES.

**Note:** If the Scan file systems control (QSCANFSCTL) value *NOPOSTRST is not specified when an object with this attribute is restored, the object will be scanned at least once after the restore.
Restricted rename and unlink (RSTDRNMUNL)

Specifies whether special restrictions apply for rename and unlink operations performed on objects within the root directory of the user-defined file system. This attribute is equivalent to the S_ISVTX mode bit for this directory.

*NO  No additional restrictions for renaming or unlinking objects from the root directory of the user-defined file system.

*YES  Objects within the root directory of the user-defined file system may be renamed or unlinked only if one or more of the following are true for the user performing the operation:
1. The user is the owner of the object.
2. The user is the owner of the directory.
3. The user has all object (*ALLOBJ) special authority.

Case sensitivity (CASE)

Specifies the case sensitivity of this file system.

*MONO  The file system will not be case sensitive. For example, the names FileA and filea refer to the same object.

*MIXED  The file system will be case sensitive. For example, the names FileA and filea do NOT refer to the same object.

Default file format (DFTFILEFMT)

Specifies the format of stream files (*STMF) created in this user-defined file system.

*TYPE2  A *TYPE2 *STMF has high performance file access and was new in Version 4 Release 4 of i5/OS. It has a minimum object size of 4096 bytes and a maximum object size of approximately 1 terabyte. A *TYPE2 stream file is capable of memory mapping as well as the ability to specify an attribute to optimize disk storage allocation.

*TYPE1  A *TYPE1 *STMF has the same format as *STMF objects created on releases prior to Version 4 Release 4 of i5/OS. It has a minimum size of 4096 bytes and a maximum object size of approximately 256 gigabytes.

Text 'description' (TEXT)

Text description for the user-defined file system.

*BLANK  Text is not specified.

character  Specify no more than 50 characters, enclosed in apostrophes.
Examples

Example 1: Create UDFS in System ASP

CRTUDFS UDFS('/dev/QASP01/joe.udfs') TEXT('Joe Smith')

This command creates a user-defined file system (UDFS) named joe.udfs in the system auxiliary storage pool (ASP 1).

Example 2: Create UDFS in ASP 3

CRTUDFS UDFS('/dev/QASP03/harry.udfs') CASE(*MIXED)

This command creates a case-sensitive user-defined file system (UDFS) named harry.udfs in user auxiliary storage pool (ASP 3).

Error messages

*ESCAPE Messages

CPFA0A2
Information passed to this operation was not valid.

CPFA09C
Not authorized to object. Object is &1.

CPFA0A9
Object not found. Object is &1.

CPFA0B1

CPFA1B8
*IOSYSCFG authority required to use &1.
Create User Profile (CRTUSRPRF)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Create User Profile (CRTUSRPRF) command identifies a user to the system and allows you to customize the way the system appears. When the profile is created, the profile is given *CHANGE and *OBJMGT authorities for the profile itself. The system relies on the profile having these authorities to itself and they should not be removed.

Restrictions: The user of this command must have:
• Security administrator (*SECADM) special authority
• Use (*USE) authority to the initial program, initial menu, job description, message queue, output queue, and attention-key-handling program (if specified)
• Change (*CHANGE) and object management (*OBJMGT) authorities to the group profile and supplemental group profiles (if specified).

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>USRPREF</td>
<td>User profile</td>
<td>Simple name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td>PASSWORD</td>
<td>User password</td>
<td>Character value, *USRPRF, *NONE</td>
<td>Optional, Positional 2</td>
</tr>
<tr>
<td>PWDEXP</td>
<td>Set password to expired</td>
<td>*NO, *YES</td>
<td>Optional</td>
</tr>
<tr>
<td>STATUS</td>
<td>Status</td>
<td>*ENABLED, *DISABLED</td>
<td>Optional</td>
</tr>
<tr>
<td>ASTLVL</td>
<td>Assistance level</td>
<td>*SYSVAL, *BASIC, *INTERMED, *ADVANCED</td>
<td>Optional</td>
</tr>
<tr>
<td>CURLIB</td>
<td>Current library</td>
<td>Name, *CRTDFT</td>
<td>Optional</td>
</tr>
<tr>
<td>INLPGM</td>
<td>Initial program to call</td>
<td>Single values: *NONE Other values: Qualified object name</td>
<td>Optional</td>
</tr>
<tr>
<td>qualifier 1: Initial program to call</td>
<td>Name</td>
<td></td>
<td></td>
</tr>
<tr>
<td>qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
<td></td>
</tr>
<tr>
<td>INLMNU</td>
<td>Initial menu</td>
<td>Single values: *SIGNOFF Other values: Qualified object name</td>
<td>Optional</td>
</tr>
<tr>
<td>qualifier 1: Initial menu</td>
<td>Name, MAIN</td>
<td></td>
<td></td>
</tr>
<tr>
<td>qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
<td></td>
</tr>
<tr>
<td>LMTCBP</td>
<td>Limit capabilities</td>
<td>*NO, *PARTIAL, *YES</td>
<td>Optional</td>
</tr>
<tr>
<td>TEXTI</td>
<td>Text ‘description’</td>
<td>Character value, *BLANK</td>
<td>Optional</td>
</tr>
<tr>
<td>SPCENV</td>
<td>Special environment</td>
<td>*SYSVAL, *NONE, *S36</td>
<td>Optional</td>
</tr>
<tr>
<td>DSPSGNINF</td>
<td>Display sign-on information</td>
<td>*SYSVAL, *NO, *YES</td>
<td>Optional</td>
</tr>
<tr>
<td>Keyword</td>
<td>Description</td>
<td>Choices</td>
<td>Notes</td>
</tr>
<tr>
<td>------------</td>
<td>----------------------------</td>
<td>---------------------------------</td>
<td>---------</td>
</tr>
<tr>
<td>PWDEXPITV</td>
<td>Password expiration interval</td>
<td>1-366, *SYSVAL, *NOMAX</td>
<td>Optional</td>
</tr>
<tr>
<td>LCLPWDMG1</td>
<td>Local password management</td>
<td>YES, *NO</td>
<td>Optional</td>
</tr>
<tr>
<td>LMTDEVSNN</td>
<td>Limit device sessions</td>
<td>*SYSVAL, *YES, *NO</td>
<td>Optional</td>
</tr>
<tr>
<td>KBDBUT</td>
<td>Keyboard buffering</td>
<td>*SYSVAL, *NO, *TYPEAhead, *YES</td>
<td>Optional</td>
</tr>
<tr>
<td>MAXSTG</td>
<td>Maximum allowed storage</td>
<td>Integer, *NOMAX</td>
<td>Optional</td>
</tr>
<tr>
<td>PTYLMT</td>
<td>Highest schedule priority</td>
<td>0-9, *NOMAX</td>
<td>Optional</td>
</tr>
<tr>
<td>JOBD</td>
<td>Job description</td>
<td>Qualifier 1: Job description</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Qualifier 2: Library</td>
<td></td>
</tr>
<tr>
<td>GRPPRF</td>
<td>Group profile</td>
<td>Name, *NONE</td>
<td>Optional</td>
</tr>
<tr>
<td>OWNER</td>
<td>Owner</td>
<td>*USRPRF, *GRPPRF</td>
<td>Optional</td>
</tr>
<tr>
<td>GRPAUTTYP</td>
<td>Group authority type</td>
<td>*PRIVATE, *PGP</td>
<td>Optional</td>
</tr>
<tr>
<td>SUPGPRRF</td>
<td>Supplemental groups</td>
<td>Single values: *NONE</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Other values (up to 15 repetitions): Name</td>
<td></td>
</tr>
<tr>
<td>ACGCDE</td>
<td>Accounting code</td>
<td>Character value, *BLANK</td>
<td>Optional</td>
</tr>
<tr>
<td>DOCPWD</td>
<td>Document password</td>
<td>Name, *NONE</td>
<td>Optional</td>
</tr>
<tr>
<td>MSGQ</td>
<td>Message queue</td>
<td>Single values: *USRPRF</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Other values: Qualified object name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Message queue</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>DLVRY</td>
<td>Delivery</td>
<td>*NOTIFY, *BREAK, *HOLD, *DFT</td>
<td>Optional</td>
</tr>
<tr>
<td>SEV</td>
<td>Severity code filter</td>
<td>0-99, 0</td>
<td>Optional</td>
</tr>
<tr>
<td>PRTDEV</td>
<td>Print device</td>
<td>Name, *WRKSTN, *SYSVAL</td>
<td>Optional</td>
</tr>
<tr>
<td>OUTQ</td>
<td>Output queue</td>
<td>Single values: *WRKSTN, *DEV</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Other values: Qualified object name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Output queue</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>ATNPGM</td>
<td>Attention program</td>
<td>Single values: *NONE, *SYSVAL, *ASSIST</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Other values: Qualified object name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Attention program</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Other values: Qualified object name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Sort sequence</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>LANGID</td>
<td>Language ID</td>
<td>Character value, *SYSVAL</td>
<td>Optional</td>
</tr>
<tr>
<td>CNTRYID</td>
<td>Country or region ID</td>
<td>Character value, *SYSVAL</td>
<td>Optional</td>
</tr>
<tr>
<td>CSSID</td>
<td>Coded character set ID</td>
<td>Integer, *SYSVAL, *HEX</td>
<td>Optional</td>
</tr>
<tr>
<td>CHRIDCTL</td>
<td>Character identifier control</td>
<td>*SYSVAL, *DEVD, *JOBCCSID</td>
<td>Optional</td>
</tr>
<tr>
<td>SETJOBATR</td>
<td>Locale job attributes</td>
<td>Single values: *SYSVAL, *NONE</td>
<td>Optional</td>
</tr>
<tr>
<td>Keyword</td>
<td>Description</td>
<td>Choices</td>
<td>Notes</td>
</tr>
<tr>
<td>-----------</td>
<td>-------------------</td>
<td>--------------------------------------------------</td>
<td>--------</td>
</tr>
<tr>
<td><strong>USROPT</strong></td>
<td>User options</td>
<td>Single values: *NONE</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>UID</strong></td>
<td>User ID number</td>
<td>1-4294967294, *GEN</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>GID</strong></td>
<td>Group ID number</td>
<td>1-4294967294, *NONE, *GEN</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>HOMEDIR</strong></td>
<td>Home directory</td>
<td>Path name, *USRPRF</td>
<td>Optional</td>
</tr>
<tr>
<td><strong>EIMASSOC</strong></td>
<td>EIM association</td>
<td>Single values: *NOCHG</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Other values: Element list</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 1: EIM identifier</td>
<td>Character value, *USRPRF</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 3: Association action</td>
<td>*REPLACE, *ADD, *REMOVE</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 4: Create EIM identifier</td>
<td>*NOCRTEIMID, *CRTEIMID</td>
<td></td>
</tr>
<tr>
<td><strong>AUT</strong></td>
<td>Authority</td>
<td>*ALL, *CHANGE, *USE, *EXCLUDE</td>
<td>Optional</td>
</tr>
</tbody>
</table>

**User profile (USRPRF)**

Specifies the user profile to be created. A numeric user profile can be specified. If the user profile is numeric, it must begin with a Q.

This is a required parameter.

**name** Specify the name of the user profile to be created.

**User password (PASSWORD)**

Specifies the password that allows the user to sign on the system. The password is associated with a user profile and is used by the system to represent the user in the system. The passwords should be known only to the individual user. A numeric password can be specified.

When the system is operating at password level 0 or 1 and the password is numeric, then the password must begin with a Q, for example, Q1234 where 1234 is the password used for signing on the system.

**Note:** The password level is controlled by the Password Level (QPWDLVL) system value.

**Note:** The new password is not checked against the password validation rules. The password validation rules are defined by i5/OS system values. For a description of the password validation rules, see the iSeries Security Reference, SC41-5302 book.

**USRPRF** The password for this user is the same as the user name specified on the USRPRF parameter. When the system is operating at password level 2 or 3 and the *USRPRF value was specified for the user profile password, the user must enter their password using upper case characters.

**NONE** No password is associated with this user profile. Users cannot sign on a system with a profile that has PASSWORD(*NONE) specified.
user-password

When the system is operating at password level 0 or 1, specify an alphanumeric character string of 10 characters or less. The first character must be alphabetic and the other characters must be alphanumeric.

When the system is operating at password level 2 or 3, specify a character string of 128 characters or less. Passwords are case sensitive at password level 2 or 3.

If the local password management (LCLPWDMGT) parameter is *NO, the local i5/OS password will be set to *NONE, so the user would have the same restrictions as specifying *NONE for the password. The password value specified will be sent to other IBM products that do password synchronization (for example, iSeries Integration for Windows Server). See the documentation for the product for information on managing the passwords for the product when LCLPWDMGT(*NO) is specified for the user profile.

Set password to expired (PWDEXP)

Specifies whether the password for this user is set to expired. If the password is set to expired, the user is required to change the password to sign on the system. When the user attempts to sign on the system, the sign-on information display is shown and the user has the option to change this password.

*NO The password is not set to expired.

*YES The password is set to expired.

Status (STATUS)

Specifies the status of the user profile.

The system will disable a user profile if the number of failed sign-on attempts reaches the limit specified on the QMAXSIGN system value and option 2 or 3 has been specified on the QMAXSGNACN system value.

*ENABLED The user profile is valid for sign-on.

*DISABLED The user profile is not valid for sign-on until an authorized user enables it again. Batch jobs can be submitted under a disabled user profile.

User class (USRCLS)

 Specifies the type of user associated with this user profile: security officer, security administrator, programmer, system operator, or user. The user class controls the options that are shown on a menu. Special authorities are given only if *USRCLS is specified for the Special authority (SPCAUT) parameter. If SPCAUT(*USRCLS) is specified, the special authorities granted will differ depending on the QSECURITY value.

*USER

At QSECURITY level 10 or 20, the user has *ALLOBJ and *SAVSYS authority.

At QSECURITY level 30 or above, the user has no special authorities.
*SECOFR
At all levels of security, the security officer is granted the following special authorities:
- *ALLOBJ
- *SAVSYS
- *JOBCTL
- *SERVICE
- *SPLCTL
- *SECADM
- *AUDIT
- *IOSYSCFG

*SECADM
At QSECURITY level 10 or 20, the security administrator has *ALLOBJ, *SAVSYS, *SECADM, and *JOBCTL special authorities.

At QSECURITY level 30 or above, the user has *SECADM special authority.

*PGMR
At QSECURITY level 10 or 20, the programmer has *ALLOBJ, *SAVSYS, and *JOBCTL special authorities.

At QSECURITY level 30 or above, the user has no special authorities.

*SYSOPR
At QSECURITY level 10 or 20, the system operator has *ALLOBJ, *SAVSYS, and *JOBCTL special authorities.

At QSECURITY level 30 or above, the user has *SAVSYS and *JOBCTL special authorities.

**Assistance level (ASTLVL)**
Specifies which user interface to use.

*SYSVAL
The assistance level defined in the system value QASTLVL is used.

*BASIC
The Operational Assistant user interface is used.

*INTERMED
The system interface is used.

*ADVANCED
The expert system interface is used. To allow for more list entries, option keys and function keys are not displayed. If a command does not have an advanced (*ADVANCED) level, the intermediate (*INTERMED) level is used.

**Current library (CURLIB)**
Specifies the name of the current library associated with the job being run.
Specifies the name of the library to be used as the current library for this user. If *PARTIAL or *YES is specified for the **Limit capabilities (LMTCPB)** parameter of the Create User Profile (CRTUSRPRF) or Change User Profile (CHGUSRPRF) command, the user cannot change the current library at sign-on or with the Change Profile (CHGPRF) command.

**CRTDFT**

This user has no current library. The library QGPL is used as the default current library.

name Specify the name of the library to use as the current library for this user.

---

**Initial program to call (INLPGM)**

Specifies, for an interactive job, the program called whenever a new routing step is started that has QCMD as the request processing program. If *PARTIAL or *YES is specified for the **Limit capabilities (LMTCPB)** parameter, the program value cannot be changed at sign on or by using the Change Profile (CHGPRF) command. No parameters can be passed to the program.

A System/36 environment procedure name can be specified as the initial program if the procedure is a member of the file QS36PRC (in the library list or specified library) and if either of the following conditions are true:

- *S36 is specified on the SPCENV parameter.
- *SYSVAL is specified on the SPCENV parameter and the system value, QSPCENV, is *S36.

**Single values**

**NONE**

No program is called when the user signs on. If a menu name is specified in the **Initial menu (INLMNU)** parameter, that menu is displayed.

**Qualifier 1: Initial program to call**

name Specify the name of the program that is called when the user signs on.

**Qualifier 2: Library**

**LIBL**

All libraries in the library list for the current thread are searched until the first match is found.

**CURLIB**

The current library for the job is used to locate the program. If no library is specified as the current library for the job, QGPL is used.

name Specify the name of the library where the initial program is located.

---

**Initial menu (INLMNU)**

Specifies the initial menu displayed when the user signs on the system if the user’s routing program is the command processor QCMD. If *YES is specified for the **Limit capabilities (LMTCPB)** parameter, the user cannot change the menu either at sign-on or with the Change Profile (CHGPRF) command.

A System/36 environment menu can be specified as the initial menu if either of the following conditions are true:

- *S36 is specified for the **Special environment (SPCENV)** parameter.
- *SYSVAL is specified on the SPCENV parameter and the system value, QSPCENV, is *S36.
Single values

**MAIN**  The menu named MAIN is located and shown.

**SIGNOFF**  
The system signs off the user when the program completes. This is intended for users authorized only to run the program.

Qualifier 1: Initial menu

`name`  Specify the name of the initial menu called after the user signs on the system.

Qualifier 2: Library

**LIBL**  All libraries in the library list for the current thread are searched until the first match is found.

**CURLIB**  
The current library for the job is used to locate the menu. If no library is specified as the current library for the job, QGPL is used.

`name`  Specify the name of the library where the initial menu is located.

Limit capabilities (LMTCPB)

Specifies the limit to which the user can control the program, menu, current library, and the ATTN key handling program values. It also determines whether the user can run commands from a command line. This parameter is ignored when the security level is 10.

**Note:** When creating or changing other users’ user profiles, you cannot specify values on this parameter that grant greater capabilities to other users than your own user profile grants to you. For example, if *PARTIAL* is specified for the Limit capabilities (LMTCPB) parameter in your user profile, you can specify *PARTIAL* or *YES* for another user. You cannot specify *NO* for another user.

**NO**  The program, menu, and current library values can be changed when the user signs on the system. Users may change the program, menu, current library, or ATTN key handling program values in their own user profiles with the Change Profile (CHGPRF) command. Commands can be run from a command line.

**PARTIAL**  
The program and current library cannot be changed on the sign-on display. The menu can be changed and commands can be run from a command line. A user can change the menu value with the Change Profile (CHGPRF) command. The program, current library, and the ATTN key handling program cannot be changed using the CHGPRF command.

**YES**  
The program, menu, and current library values cannot be changed on the sign-on display. Commands cannot be run when issued from a command line or by selecting an option from a command grouping menu such as CMDADD, but can still be run from a command entry screen. The user cannot change the program, menu, current library, or the ATTN key program handling values by using the CHGPRF command.

Text 'description' (TEXT)

Specifies the text that briefly describes the object.

**BLANK**  
No text is specified.
'description'
  Specify no more than 50 characters of text, enclosed in apostrophes.

---

### Special authority (SPCAUT)

Specifies the special authorities given to a user. Special authorities are required to perform certain functions on the system. Special authorities cannot be removed from many of the system-supplied user profiles, including QSECOFR and QSYS.

The following special authorities are usually given:

- Save system (*SAVSYS) special authority to users who need to operate the system.
- Input/output system configuration (*IOSYSCFG) special authority to users who need to change system I/O configurations.
- Job control (*JOBCTL) special authority is given to the user. The user is given the authority to change, display, hold, release, cancel, and clear all jobs that are running on the system or that are on a job queue or output queue that has OPRCTL (*YES) specified. The user also has the authority to load the system, to start writers, and to stop active subsystems.
- Security administrator (*SECADM) special authority to users who need to create, change, or delete user profiles.
- All object (*ALLOBJ) special authority to users who need to work with system resources.
- Service (*SERVICE) special authority to users who need to perform service functions.
- Spool control (*SPLCTL) special authority to users who need to perform all spool-related functions.
- Audit (*AUDIT) special authority to users who need to perform auditing functions.

**Restrictions:**

- The user profile creating or changing another user profile must have all of the special authorities being given. All special authorities are needed to give all special authorities to another user profile.
- A user must have *ALLOBJ and *SECADM special authorities to give a user *SECADM special authority when using the CHGUSRPRF command.
- The user must have *ALLOBJ, *SECADM, and *AUDIT special authorities to give a user *AUDIT special authority when using the CHGUSRPRF command.

**Single values**

**USRCLS**

Special authorities are granted to this user based on the value specified on User class (USRCLS) parameter.

**NONE**

No special authorities are granted to this user.

**Other values**

**ALLOBJ**

All object authority is given to the user. The user can access any system resource with or without private user authorizations.

**AUDIT**

Audit authority is granted to this user. The user is given the authority to perform auditing functions. Auditing functions include turning auditing on or off for the system and controlling the level of auditing on an object or user.
*JOBCTL
Job control authority is given to the user. The user has authority to change, display, hold, release, cancel, and clear all jobs that are running on the system or that are on a job queue or output queue that has OPRCTL (*YES) specified. The user also has the authority to start writers and to stop active subsystems.

*SAVSYS
Save system authority is given to the user profile. This user has the authority to save, restore, and free storage for all objects on the system, with or without object management authority.

*IOSYSCFG
Input/output (I/O) system configuration authority is given to the user. The user has authority to change system I/O configurations.

*SECADM
Security administrator authority is given to the user. The user can create, change, or delete user profiles if authorized to the Create User Profile (CRTUSRPRF), Change User Profile (CHGUSRPRF), or Delete User Profile (DLTUSRPRF) commands and is authorized to the user profile. This authority does not allow giving special authorities that this user profile does not have. To give *SECADM special authority to another user, a user must have both *ALLOBJ and *SECADM special authorities.

*SERVICE
Service authority is given to this user. The user can perform service functions.

*SPLCTL
Spool control authority is given to this user. The user can perform all spool functions.

Special environment (SPCENV)
Specifies the special environment in which the user operates after signing on.

*SYSVAL
The system value, QSPCENV, is used to determine the system environment after the user signs on the system.

*NONE
The user operates in the i5/OS system environment after signing on the system.

*S36
The user operates in the System/36 environment after signing on the system.

Display sign-on information (DSPSGNINF)
Specifies whether the sign-on information display is shown.

*SYSVAL
The system value QDSPSGNINF is used to determine whether the sign-on information display is shown.

*NO
The sign-on information display is not shown.

*YES
The sign-on information display is shown.
Password expiration interval (PWDEXPIITV)

Specifies the password expiration interval (in days).

*SYSVAL
   The system value QPWDEXPIITV is used to determine the password expiration interval.

*NOMAX
   The password does not expire.

1-366 Specify the number of days between the date when the password is changed and the date when the password expires. Valid values range from 1 through 366.

Local password management (LCLPWDMGT)

Specifies whether the user profile password should be managed locally.

*YES Password will be managed on the local system.

*NO Password will not be managed on the local system. Specifying this value will cause the local i5/OS password to be set to *NONE. The password value specified in the password parameter will be sent to other IBM products that do password synchronization (for example, iSeries Integration for Windows Server).

   The user will not be able to change their own password using the Change Password (CHGPWD) command. They also will not be able to sign on to the system directly.

   Specifying this value will affect other IBM products that do password synchronization, like iSeries Integration for Windows Server. See the documentation for the product for details.

   This value should be used if the user only needs to access the system through some other platform, such as Windows.

Limit device sessions (LMTDEVSSN)

Specifies if the number of device sessions allowed for a user is limited to 1. This does not limit SYSREQ and second sign-on.

*SYSVAL
   The system value QLMTDEVSSN is used to determine whether the user is limited to a single device session.

*NO The user is not limited to a single device session.

*YES The user is limited to a single device session.

Keyboard buffering (KBDBUF)

Specifies the keyboard buffering value to be used when a job is initialized for this user profile. If the type-ahead feature is active, you can buffer your keyboard strokes. If the attention key buffering option is active, the attention key is buffered as any other key. If it is not active, the attention key is not buffered and is sent to the system even if the display station is input-inhibited. This value can also be set by a user application. More information is in the System API Reference information in the iSeries Information Center at http://www.ibm.com/eserver/iseries/infocenter.

394 IBM Systems - iSeries: i5/OS Commands Starting with CRTMSGF (Create Message File)
*SYSVAL
   The system value, QKBDBUF, is used to determine the keyboard buffering value.

*NO   The type-ahead feature and attention key buffering option are not active.

*TYPEAHEAD   The type-ahead feature is active, but the attention key buffering option is not.

*YES   The type-ahead feature and attention key buffering option are active.

---

**Maximum allowed storage (MAXSTG)**

Specifies the maximum amount of auxiliary storage (in kilobytes) assigned to store permanent objects owned by this user profile (1 kilobyte equals 1024 bytes). If the maximum is exceeded when an interactive user tries to create an object, an error message is displayed, and the object is not created. If the maximum is exceeded when an object is created in a batch job, an error message is sent to the job log (depending on the logging level of the job), and the object is not created.

Storage is allocated in 4K increments. Therefore, if you specify MAXSTG (9), the profile is allocated 12K of storage.

When planning maximum storage for user profiles, consider the following system actions:

- A restore operation assigns the storage to the user doing the restore, and then transfers the object to the owner. For a large restore, specify MAXSTG(*NOMAX).
- The user profile that creates a journal receiver is assigned the required storage as the receiver size grows. If new receivers are created using JRNRCV(*GEN), the storage continues to be assigned to the user profile that owns the active journal receiver. If a very active journal receiver is owned, specify MAXSTG(*NOMAX).
- User profiles that transfer created objects to their group profile must have adequate storage in the user profiles to contain created objects before the objects are transferred to the group profile.
- The owner of the library is assigned the storage for the descriptions of objects which are stored in a library, even when the objects are owned by another user profile. Examples of such objects are text and program references.

*NOMAX   As much storage as is required is assigned to this profile.

**number**

Specify the maximum amount of storage for the user, in kilobytes (1 kilobyte equals 1024 bytes).

---

**Highest schedule priority (PTYLMT)**

Specifies the highest scheduling priority the user is allowed to have for each job submitted to the system. This value controls the job processing priority and output priority for any job running under this user profile; that is, values specified in the JOBPTY and OUTPTY parameters of any job command cannot exceed the PTYLMT value of the user profile under which the job is run. The scheduling priority can have a value ranging from 0 through 9, where 0 is the highest priority and 9 is the lowest priority.

**3**   The user named in this profile can have a priority value no higher than 3 for scheduling jobs on the system.

**0-9**  Specify a value ranging from 0 through 9 for the highest scheduling priority that the user is allowed.

Create User Profile (CRTUSRPRF)   395
Job description (JOBD)

Specifies the job description used for jobs that start through subsystem work station entries. If the job description does not exist when the user profile is created or changed, a library qualifier must be specified, because the job description name is kept in the user profile.

Qualifier 1: Job description

QDFTJOBD

The default system-supplied job description found in library QGPL is used.

name Specify the name of job description used for the work station entries whose job description parameter values indicate the user JOBD(USRPRF).

Qualifier 2: Library

*LIBL All libraries in the library list for the current thread are searched until the first match is found.

*CURLIB

The current library for the thread is searched. If no library is specified as the current library for the thread, the QGPL library is used.

name Specify the name of the library to be searched.

Group profile (GRPPRF)

Specifies the user’s group profile name whose authority is used if no specific authority is given for the user. The current user of this command must have object management (*OBJMGT) and change (*CHANGE) authority to the profile specified for the Group profile (GRPPRF) parameter. The required *OBJMGT authority cannot be given by a program adopt operation.

Note:
1. When a group profile is specified, the user is automatically granted *CHANGE and *OBJMGT authority to the group profile.
2. The following IBM-supplied objects are not valid on this parameter.
   QANZAGENT, QAUTPROF, QCOLUMN, QCOLUMN, QCCLUSTER, QCOLSRV, QDBSRV, QDBSRVDO,
   QDFTOWN, QDIRSRV, QDLMF, QDOC, QDNSX, QEJB, QFNC, QGATE, QIBMHELP, QIPP,
   QLPAUTO, QLPINSTALL, QMGTC, QMSF, QNETSFLE, QNFSANON, QNTP, QPEX, QPM400, QRIE,
   QSNADS, QSPL, QSPLJOB, QSRL, QSRVAGT, QSRVBAS, QSYS, QTCM, QTCP, QTFTP, QTSTRQS,
   QYCMICIMOM, QYPSJVR

*NONE This user profile has no group profile.

name Specify the name of the group profile used with this user profile.

Owner (OWNER)

Specifies the user profile that is to be the owner of objects created by this user.
The user profile associated with the job is the owner of the object.

The group profile is made the owner of newly created objects and has all authority to the object. The user profile associated with the job does not have any specific authority to the object. If *GRPPRF is specified, a user profile name must be specified for the Group profile (GRPPRF) parameter, and the Group authority (GRPAUT) parameter cannot be specified.

Group authority (GRPAUT)

The specific authority given to the group profile for newly created objects. If *GRPPRF is specified for the Owner (OWNER) parameter, specification of this parameter is not allowed.

*NONE
No group authority is given.

*ALL
The user can perform all operations except those limited to the owner or controlled by authorization list management (*AUTLMGT) authority. The user can control the object’s existence, specify the security for the object, change the object, and perform basic functions on the object. The user also can change ownership of the object.

*CHANGE
The user can perform all operations on the object except those limited to the owner or controlled by object existence (*OBJEXIST) and object management (*OBJMGT) authorities. The user can change and perform basic functions on the object. *CHANGE authority provides object operational (*OBJOPR) authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove users.

*USE
The user can perform basic operations on the object, such as running a program or reading a file. The user cannot change the object. Use (*USE) authority provides object operational (*OBJOPR), read (*READ), and execute (*EXECUTE) authorities.

*EXCLUDE
The user cannot access the object.

Group authority type (GRPAUTTYP)

Specifies the type of authority to be granted to the group profile for newly-created objects. If *NONE is specified for the Group authority (GRPAUT) parameter, specification of this parameter is ignored.

*PRIVATE
The group profile is granted private authority to newly-created objects, with the authority value determined by the GRPAUT parameter. If the authority value in the GRPAUT parameter is *NONE, this value is ignored.

*PGP
The group profile is be the primary group for newly-created objects, with the authority value determined by the GRPAUT parameter. If the authority value in the GRPAUT parameter is *NONE, this value is ignored.
Supplemental groups (SUPGRPPRF)

Specifies the user’s supplemental group profiles. The profiles specified here, along with the group profile specified for the Group profile (GRPPRF) parameter, are used to determine what authority the user has if no specific user authority is given for the job. If profiles are specified for this parameter, a group profile name must be specified on the GRPPRF parameter for this user profile (either on this command or on a previous Create User Profile (CRTUSRPRF) or Change User Profile (CHGUSRPRF) command. The current user of this command must have object management (*OBJMGT) and change (*CHANGE) authority to the profiles specified for this. The required *OBJMGT authority cannot be given by a program adopt operation.

Notes:
1. When a group profile is specified, the user is automatically granted *CHANGE and *OBJMGT authority to the group profile.
2. The following IBM-supplied user profiles are not valid for this parameter:
   - QANZAGENT, QAUTPROF, QCUMGMT, QCCLUSTER, QCOLSRV, QDBSRH, QDBSRHDO, QDFTOWN, QDIRSRV, QDLFM, QDOC, QDSNX, QEJB, QFNC, QGATE, QIBMHELP, QIPP, QLPAUTO, QLPINSTALL, QMGT, QMSF, QNETSPLF, QNFSANON, QNTP, QPEX, QPM400, QRIE, QSNADS, QSLP, QSQLJOB, QSRV, QSRVAGT, QSRVBAS, QSYS, QTCM, QTCP, QTFTP, QTSTRQS, QYCMCIMOM, QYPJSVR

   *NONE
      No supplemental group profiles are used with this user profile.

   name
      Specify a maximum of 15 group profile names used with this user profile and the group profile specified on the GRPPRF parameter to determine a job’s eligibility for getting access to existing objects and special authority.

Accounting code (ACGCDE)

Specifies the accounting code that is associated with this user profile.

   *BLANK
      An accounting code consisting of 15 blanks is assigned to this user profile.

   character-value
      Specify the 15-character accounting code to be used by jobs that get their accounting code from this user profile. If less than 15 characters are specified, the string is padded on the right with blanks.

Document password (DOCPWD)

Specifies the document password that allows Document Interchange Architecture (DIA) document distribution services users protect personal distributions from being used by people who work on their behalf.

   *NONE
      No document password is used by this user.

   name
      Specify the document password to be assigned to this user. The password must range from 1 through 8 alphanumeric characters (letters A through Z and numbers 0 through 9). The first character of the document password must be alphabetic; the remaining characters can be alphanumeric. Embedded blanks, leading blanks, and special characters are not valid.
**Message queue (MSGQ)**

Specifies the message queue to which messages are sent.

**Note:** The message queue is created, if it does not already exist. The user profile specified for the **User profile (USRPRF)** parameter is the owner of the message queue.

**Single values**

*USRPRF*  
A message queue with the same name as that specified for the USRPRF parameter is used as the message queue for this user. This message queue is located in the QUSRSYS library.

**Qualifier 1: Message queue**

*name*  
Specify the name of the message queue to be used with this profile.

**Qualifier 2: Library**

*LIBL*  
All libraries in the library list for the current thread are searched until the first match is found.

*CURLIB*  
The current library for the thread is searched. If no library is specified as the current library for the thread, the QGPL library is used.

*name*  
Specify the name of the library to be searched.

---

**Delivery (DLVRY)**

Specifies how messages are sent to the message queue for this user are to be delivered.

*NOTIFY*  
The job to which the message queue is assigned is notified when a message arrives at the message queue. For interactive jobs at a work station, the audible alarm is sounded (if the alarm feature is set) and the Message Waiting light is turned on. The delivery mode cannot be changed to *NOTIFY if the message queue is also being used by another job.

*HOLD*  
The messages are held in the message queue until they are requested by the user or program.

*BREAK*  
The job to which the message queue is assigned is interrupted when a message arrives at the message queue. If the job is an interactive job, the audible alarm is sounded (if the alarm feature is set). The delivery mode cannot be changed to *BREAK if the message queue is also being used by another job.

*DFT*  
The default reply to the inquiry message is sent. If no default reply is specified in the message description of the inquiry message, the system default reply, *N, is used.
Severity code filter (SEV)

Specifies the lowest severity code that a message can have and still be delivered to a user in break or notify mode. Messages arriving at the message queue whose severities are lower than the severity code specified for this parameter do not interrupt the job or turn on the audible alarm or the message-waiting light; they are held in the queue until they are requested by using the Display Message (DSPMSG) command. If *BREAK or *NOTIFY is specified for the Delivery (DLVRY) parameter, and is in effect when a message arrives at the queue, the message is delivered if the severity code associated with the message is equal or greater than the value specified here. Otherwise, the message is held in the queue until it is requested.

0 If a severity code is not specified, 0 is used.
0-99 Specify a severity code ranging from 00 through 99.

Print device (PRTDEV)

Specifies the default printer device for this user. If the printer file used to create printed output specifies to spool the data, the spooled file is placed on the device’s output queue, which is named the same as the device.

Note: This assumes the defaults are specified for the Output queue (OUTQ) parameter for the printer file, job description, user profile and workstation.

*WRKSTN The printer assigned to the user’s work station is used.

*SYSVAL The value specified in the system value QPRTDEV is used.

name Specify the name of a printer that is to be used to print the output for this user.

Output queue (OUTQ)

Specifies the output queue to be used by this user profile. The output queue must already exist when this command is run.

Single values

*WRKSTN The output queue assigned to the user’s work station is used.

*DEV The output queue associated with the printer specified for the Print device (PRTDEV) parameter is used. The output queue has the same name as the printer. (The printer file DEV parameter is determined by the CRTPRTF, CHGPRTF, or the OVRPRTF command).

Note: This assumes the defaults are specified for the Output queue (OUTQ) parameter for the printer file, job description, user profile and workstation.

Qualifier 1: Output queue

name Specify the name of the output queue to be used by this user profile.

Qualifier 2: Library

*LIBL All libraries in the library list for the current thread are searched until the first match is found.
*CURLIB
   The current library for the thread is searched. If no library is specified as the current library for the thread, the QGPL library is used.

name   Specify the name of the library to be searched.

---

**Attention program (ATNPGM)**

Specifies the program to be used as the Attention (ATTN) key handling program for this user. The ATTN key handling program is called when the ATTN key is pressed during an interactive job. The program is active only when the user routes to the system-supplied QCMD command processor. The ATTN key handling program is set on before the initial program (if any) is called and it is active for both program and menu. If the program changes the ATNPGM (by using the SETATNPGM command), the new program remains active only for the duration of the program. When control returns and QCMD calls the menu, the original ATTN key handling program becomes active again. If the SETATNPGM command is run from the menus or an application is called from the menus, the new ATTN key handling program that is specified overrides the original ATTN key handling program. If *YES or *PARTIAL is specified for the Limit capabilities (LMTCPB) parameter on the Create User Profile (CRTUSRPRF) or Change User Profile (CHGUSRPRF) command, the ATTN key handling program cannot be changed.

**Single values**

*SYSVAL
   The system value QATNPGM is used.

*NONE
   No ATTN key handling program is used by this user.

*ASSIST
   The Operational Assistant ATTN key handling program, QEZMAIN, is used.

**Qualifier 1: Attention program**

name   Specifies the name of the ATTN key handling program to be used for this user profile.

**Qualifier 2: Library**

*LIBL
   All libraries in the library list for the current thread are searched until the first match is found.

*CURLIB
   The current library for the thread is searched. If no library is specified as the current library for the thread, the QGPL library is used.

name   Specify the name of the library to be searched.

---

**Sort sequence (SRTSEQ)**

Specifies the sort sequence table to be used for string comparisons for this profile.

**Single values**

*SYSVAL
   The system value QSRTSEQ is used.

*HEX
   A sort sequence table is not used. The hexadecimal values of the characters are used to determine the sort sequence.
*LANGIDUNQ
   A unique-weight sort table is used.

*LANGIDSHR
   A shared-weight sort table is used.

Qualifier 1: Sort sequence

name Specify the name of the sort sequence table to be used with this profile.

Qualifier 2: Library

*LIBL All libraries in the library list for the current thread are searched until the first match is found.

*CURLIB The current library for the thread is searched. If no library is specified as the current library for the thread, the QGPL library is used.

name Specify the name of the library to be searched.

Language ID (LANGID)

Specifies the language identifier to be used for this user.

*SYSVAL The system value QLANGID is used.

language-identifier Specify the language identifier to be used. More information on valid language identifiers is in the Globalization topic in the iSeries Information Center at http://www.ibm.com/eserver/iseries/infocenter.

Country or region ID (CNTRYID)

Specifies the country or region identifier to be used for this user.

*SYSVAL The system value QCNTRYID is used.

character-value Specify a country or region identifier. To see a complete list of identifiers when prompting this command, position the cursor on the field for this parameter and press F4 (Prompt).

Coded character set ID (CCSID)

Specifies the coded character set identifier (CCSID) to be used for this user.

A CCSID is a 16-bit number identifying a specific set of encoding scheme identifiers, character set identifiers, code page identifiers, and additional coding-related information that uniquely identifies the coded graphic representation used.

Note: If the value for CCSID is changed, the change does not affect jobs that are currently running.
**SYSVAL**
The system value QCCSID is used.

**HEX**  The CCSID 65535 is used.

*identifier*
 Specify the CCSID to be used for this user profile. More information on valid CCSIDs is in the Globalization information in the iSeries Information Center at http://www.ibm.com/eserver/iseries/infoCenter.

---

**Character identifier control (CHRIDCTL)**

Specifies the character identifier control (CHRIDCTL) for the job. This attribute controls the type of coded character set identifier (CCSID) conversion that occurs for display files, printer files and panel groups. The *CHRIDCTL special value must be specified for the Character identifier (CHRID) parameter on the create, change, or override commands for display files, printer files, and panel groups before this attribute will be used.

**SYSVAL**
The system value QCHRIDCTL is used.

**DEVD**
The *DEVD special value performs the same function as on the CHRID command parameter for display files, printer files, and panel groups.

**JOBCCSID**
The *JOBCCSID special value performs the same function as on the CHRID command parameter for display files, printer files, and panel groups.

---

**Locale job attributes (SETJOBATR)**

Specifies which job attributes are to be taken from the locale specified for the Locale (LOCALE) parameter when the job is initiated.

**Single values**

**SYSVAL**
The system value, QSETJOBATR, is used to determine which job attributes are taken from the locale.

**NONE**
No job attributes are taken from the locale.

**Other values**

**CCSID**
The coded character set identifier from the locale is used. The CCSID value from the locale overrides the user profile CCSID.

**DATFMT**
The date format from the locale is used.

**DATSEP**
The date separator from the locale is used.
**DECFMT**
   The decimal format from the locale is used.

**SRTSEQ**
   The sort sequence from the locale is used. The sort sequence from the locale overrides the user profile sort sequence.

**TIMSEP**
   The time separator from the locale is used.

---

**Locale (LOCALE)**

Specifies the path name of the locale that is assigned to the LANG environment variable for this user.

**SYSVAL**
   The system value QLOCALE is used to determine the locale path name to be assigned for this user.

**NONE**
   No locale path name is assigned for this user.

**C**
   The C locale path name is assigned for this user.

**POSIX**
   The POSIX locale path name is assigned for this user.

`path-name`
   Specify the path name of the locale to be assigned for this user.

---

**User options (USROPT)**

Specifies the level of help information detail to be shown and the function of the Page Up and Page Down keys by default. The system shows several displays that are suitable for the inexperienced user. More experienced users must perform an extra action to see detailed information. When values are specified for this parameter, the system presents detailed information without further action by the experienced user.

**Single values**

**NONE**
   Detailed information is not shown.

**Other values**

**CLKWD**
   Parameter keywords are shown instead of the possible parameter values when a control language (CL) command is prompted.

**EXPERT**
   More detailed information is shown when the user is performing display and edit options to define or change the system (such as edit or display object authority).

**ROLLKEY**
   The actions of the Page Up and Page Down keys are reversed.

**NOSTSMMSG**
   Status messages are not displayed when sent to the user.
*STSMSG
Status messages are displayed when sent to the user.

*HLPFULL
Help text is shown on a full display rather than in a window.

*PRTMSG
A message is sent to this user’s message queue when a spooled file for this user is printed or held by the printer writer.

---

**User ID number (UID)**

Specifies the user ID number (uid number) for this user profile. The uid number is used to identify the user when the user is using the directory file system. The uid number for a user cannot be changed if there are one or more active jobs for the user.

*GEN
The uid number is generated for the user. The system generates a uid number that is not already assigned to another user. The uid number generated is greater than 100.

number
Specify the uid number to be assigned to the user profile. A value from 1 to 4294967294 can be entered. The uid number assigned must not already be assigned to another user profile.

---

**Group ID number (GID)**

Specify the group ID number (gid number) for this user profile. The gid number is used to identify the group profile when a member of the group is using the directory file system. The gid number for a user may not be changed if:

- The user profile is the primary group of an object in a directory.
- There are one or more active jobs for the user.

*NONE
The user does not have a gid number or an existing gid number is removed.

*GEN
The gid number will be generated for the user. The system generates a gid number that is not already assigned to another user. The gid number generated is greater than 100.

number
Specify the gid number to be assigned to the user profile. A value from 1 to 4294967294 can be entered. The gid number assigned must not already be assigned to another user profile.

---

**Home directory (HOMEDIR)**

Specifies the path name of the home directory for this user profile. The home directory is the user’s initial working directory. The working directory, associated with a process, is used during path name resolution in the directory file system for path names that do not begin with a slash (/). If the home directory specified does not exist when the user signs on, the user’s initial working directory is the root (/) directory.

*USRPRF
The home directory assigned to the user will be /home/USRPRF, where USRPRF is the name of the user profile.
'path-name'  
Specify the path name of the home directory to be assigned to this user.

For more information on specifying path names, refer to "Object naming rules" in the CL concepts and reference topic in the iSeries Information Center at http://www.ibm.com/eserver/iseries/infocenter.

---

**EIM association (EIMASSOC)**

Specifies whether an EIM (Enterprise Identity Mapping) association should be added to an EIM identifier for this user.

**Note.**
1. This information is not stored in the user profile. This information is not saved or restored with the user profile.
2. If this system is not configured for EIM, then no processing is done. Not being able to perform EIM operations does not cause the command to fail.

**Single values**

*NOCHG  
EIM association will not be added.

**Element 1: EIM identifier**

Specifies the EIM identifier for this association.

*USRPRF  
The name of the EIM identifier is the same name as the user profile.

*character-value  
Specify the name of the EIM identifier.

**Element 2: Association type**

Specifies the type of association. It is recommended that a target association is added for an i5/OS user.

Target associations are primarily used to secure existing data. They will be found as the result of a mapping lookup operation (that is, eimGetTargetFromSource()), but cannot be used as the source identity for a mapping lookup operation.

Source associations are primarily for authentication purposes. They can be used as the source identity of a mapping lookup operation, but will not be found as the target of a mapping lookup operation.

Administrative associations are used to show that an identity is associated with an EIM identifier, but cannot be used as the source for, and will not be found as the target of, a mapping lookup operation.

*TARGET  
Process a target association.

*SOURCE  
Process a source association.

*TGTSRC  
Process both a target and a source association.
**ADMIN**
Process an administrative association.

**ALL**  Process all association types.

Element 3: Association action

**REPLACE**
Associations of the specified type will be removed from all EIM identifiers that have an association for this user profile and local EIM registry. A new association will be added to the specified EIM identifier.

**ADD**  Add an association.

**REMOVE**
Remove an association.

Element 4: Create EIM identifier

Specifies whether the EIM identifier should be created if it does not already exist.

**NOCRTEIMID**
EIM identifier does not get created.

**CRTEIMID**
EIM identifier gets created if it does not exist.

**Authority (AUT)**

Specifies the authority you are giving to users who do not have specific authority for the object, who are not on an authorization list, and whose group profile or supplemental group profiles do not have specific authority for the object.

**EXCLUDE**
The user cannot access the object.

**ALL**  The user performs all operations on the object except those limited to the owner.

**CHANGE**
The user can perform all operations on the object except those limited to the owner or controlled by object existence (*OBJEXIST) and object management (*OBJMGT) authorities. The user can change and perform basic functions on the object. *CHANGE authority provides object operational (*OBJOPR) authority and all data authority. If the object is an authorization list, the user cannot add, change, or remove users.

**USE**
The user can perform basic operations on the object, such as running a program or reading a file. The user cannot change the object. Use (*USE) authority provides object operational (*OBJOPR), read (*READ), and execute (*EXECUTE) authorities.

**Examples**

Example 1: Creating a User Profile

```plaintext
CRTUSRPRF USRPRF(JJADAMS) PASSWORD(S1CR2T) SPCAUT(*SAVSYS)
INLPGM(ARLIB/DSPMENU)
```
This command creates a user profile with the user name of JJADAMS and a password of S1CR2T. After sign-on, a program called DSPMENU in the ARLIB library is called. The user is granted the save system special authority. Because the other parameters were not specified: (1) The profile has no limit on the amount of storage assigned to it for owned permanent objects; (2) A scheduling priority of 3 is the highest priority that any of the user's jobs can have; (3) The user-defined description text is blank; (4) There is no group profile associated with this user profile; and (5) No authority is granted for the user profile to other users.

Example 2: Creating a User Profile with the Same User Name and Password

```
crtusrprf usprf(tmsmith) maxstg(12) inlgm(progmr/calc)
   text('Ted Smith, Dept 410, Application Programs')
```

This command creates a user profile with the user name of TMSMITH; the password is also TMSMITH because the password was not specified. The maximum permanent storage space the user can use for all objects is 12K (or 12,288 bytes). The initial program called following sign-on is CALC, which is located in the library named PROGMR. The text parameter provides the user's name, department, and department name. Default values are assigned to the other parameters.

**Error messages**

*ESCAPE Messages*

CPF22CE
The &1 value &2 is used by another user profile.

CPF22CF
User profile not allowed to be a group profile.

CPF22DB
The user profile being changed must have a GID.

CPF22DF
Unable to process request for user profile &1.

CPF22EB
Unable to process request for user profile &1.

CPF22E1
USRLOPT parameter cannot specify *STSM SG and *NOSTSM SG.

CPF22F1
Coded character set identifier &1 not valid.

CPF22F3
&1 specified a LM TCPB value that is not permitted.

CPF22F5
Value for new password not allowed at password level &2.

CPF2202
Do not have authority to create user profile.

CPF2209
Library &1 not found.

CPF2213
Not able to allocate user profile &1.

CPF2214
User profile &1 already exists.
CPF2225
Not able to allocate internal system object.

CPF224A
User profile &1 cannot have a GID and be a member of a group.

CPF2242
Object &1 type *&2 not found in library list.

CPF2244
Object &1 type *&2 cannot be found.

CPF225A
User profile name specified on both USRPRF and SUPGRPPRF parameters.

CPF2259
Group profile &1 not found.

CPF2260
User profile &2 was not created or changed. Reason code &3.

CPF2261
OWNER or GRPAUT value not permitted.

CPF2262
Value for GRPAUT not correct.

CPF2269
Special authority *ALLOBJ required when granting *SECADM or *AUDIT.

CPF2272
Cannot allocate user profile &1.

CPF2291
User profile does not have all special authorities being granted.

CPF2292
*SECADM required to create or change user profiles.

CPF2293
Storage limit exceeded for user profile &1.

CPF9802
Not authorized to object &2 in &3.

CPF9820
Not authorized to use library &1.

CPF9825
Not authorized to device &1.
Create Validation List (CRTVLDL)

Where allowed to run: All environments (*ALL)
Threadsafe: Yes

The Create Validation List (CRTVLDL) command creates a validation list. Validation lists contain entries consisting of an identifier, data that will be encrypted by the system when it is stored, and free-form data. Entries can be added, changed, removed, found, and validated. Entries are validated by providing the correct entry identifier and data that is encrypted. See the System API Reference information in the iSeries Information Center at http://www.ibm.com/eserver/iseries/infocenter for information on how to use validation lists.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>VLDL</td>
<td>Validation list</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Validation list</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>TEXT</td>
<td>Text ‘description’</td>
<td>Character value, *BLANK</td>
<td>Optional</td>
</tr>
<tr>
<td>AUT</td>
<td>Authority</td>
<td>Name, *EXCLUDE, *USE, *CHANGE, *ALL</td>
<td>Optional</td>
</tr>
</tbody>
</table>

Validation list (VLDL)

Specifies the validation list to be created.

This is a required parameter.

Qualifier 1: Validation list

name Specify the name to be given to the validation list object.

Qualifier 2: Library

*CURLIB
The validation list is created in the current library. If no library is specified as the current library for the job, QGPL is used.

name Specify the name of the library where the validation list is created.

Text ’description’ (TEXT)

Specifies the text that briefly describes the validation list to be created.

*BLANK
No text is specified.
**Authority (AUT)**

The authority you are giving to users who do not have specific authority to the object, who are not on an authorization list, and whose group profile or supplemental group profiles have no specific authority to the object.

*EXCLUDE

Exclude authority prevents the user from accessing the object.

*USE

Use authority allows the user to perform finds and validate entries in the validation list.

*CHANGE

Change authority allows the user to add, change, remove, find, and validate entries in the validation list.

*ALL

All authority allows the user to add, change, remove, find, and validation entries in the validation list, and delete the validation list.

**name**

The name of an authorization list to be used for authority to the object. Users included in the authorization list are granted authority to the object as specified in the list. The authorization list must exist when the object is created.

**Examples**

```
CRTVLDL VLDL(WEBLIB/WEBUSRS) AUT(*EXCLUDE)
TEXT('My WEB users')
```

This command creates a validation list (WEBUSRS) in the WEBLIB library.

**Error messages**

*ESCAPE Messages*

CPF2108

Object &1 type *&3 not added to library &2.

CPF2112

Object &1 in &2 type *&3 already exists.

CPF2113

Cannot allocate library &1.

CPF2151

Operation failed for &2 in &1 type *&3.

CPF2182

Not authorized to library &1.

CPF2183

Object &1 cannot be moved into library &3.
CPF9802
   Not authorized to object &2 in &3.
CPF9803
   Cannot allocate object &2 in library &3.
CPF9810
   Library &1 not found.
CPF9818
   Object &2 in library &3 not created.
CPF9819
   Object &2 in library &3 not created.
Create WSCST (CRTWSCST)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Create Work Station Customizing Object (CRTWSCST) command allows the user to create a work station customizing object in a library.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>WSCST</td>
<td>Workstation customizing object</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Workstation customizing object</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>SRCMBR</td>
<td>Source member</td>
<td>Name, *WSCST</td>
<td>Optional, Positional 3</td>
</tr>
<tr>
<td>TEXT</td>
<td>Text ‘description’</td>
<td>Character value, *BLANK, *SRCMBRTXT</td>
<td>Optional</td>
</tr>
<tr>
<td>SRCFILE</td>
<td>Source file</td>
<td>Qualified object name</td>
<td>Optional, Positional 2</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Source file</td>
<td>Name, QTXTSRC</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>REPLACE</td>
<td>Replace object</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
</tbody>
</table>

Workstation customizing object (WSCST)

Specifies the name of a work station customizing object to be created.

The possible library values are:

*CURLIB

The work station customizing object is created in the current library for the job. If no library is specified as the current library for the job, the QGPL library is used.

library-name

Specify the name of the library where the work station customizing object is created.

The possible values are:

object-name

Specify the name of the object that is created.
Source member (SRCMBR)

Specifies the name of the source file member containing the table attributes.

The possible values are:

*WSCST
  Specifies that the work station customizing object name is used as the source member name.

source-file-member-name
  Specify the name of the member in the source file specified on the SRCFILE parameter that is used to create the work station customizing object.

Text ’description’ (TEXT)

Specifies text that briefly describes the object. More information on this parameter is in "Appendix A, Expanded Parameter Descriptions" in the CL Reference.

The possible values are:

*SRCMBRTXT
  The text is taken from the source file member used to create the work station customizing object.

*BLANK
  Text is not specified.

’description’
  Specify a description of the object.

Source file (SRCFILE)

Specifies the name of the source file in which a source file member containing the customizing table attributes resides. If the source file does not exist, an error message is displayed. The coded character set identifier for the source file should be *HEX.

The possible library values are:

*CURLIB
  The current library for the job is used to locate the source file. If no library is specified as the current library for the job, the QGPL library is used.

library-name
  Specify the name of the library where the source file is located.

The possible values are:

QTXTSRC
  The IBM-supplied source file QTXTSRC is used.

source-file-name
  Specify the name of the source file.
**Authority (AUT)**

Specifies the authority given to users who do not have specific authority to the object, who are not on the authorization list, and whose user group has no specific authority to the object.

The possible values are:

* **LIBCRTAUT**
  The public authority for the object is taken from the CRTAUT value of the target library (the library that is to contain the object). This value is determined when the object is created. If the CRTAUT value for the library changes after the object is created, the new value does not affect any existing objects.

* **CHANGE**
  The user can perform all operations on the object except those limited to the owner or controlled by object existence authority and object management authority. The user can change and perform basic functions on the object. Change authority provides object operational authority and all data authority.

* **ALL**
  The user can perform all operations except those limited to the owner or controlled by authorization list management authority. The user can control the file’s existence, specify the security for the file, change the file, and perform basic functions on the file. The user can transfer ownership of the file.

* **USE**
  The user can perform basic operations, such as run a program or read a file. The user is prevented from changing the object. *USE authority provides object operational authority, read authority and execute authority.

* **EXCLUDE**
  The user is prevented from accessing the object.

**Replace object (REPLACE)**

Indicates whether an existing object is replaced.

The possible values are:

* **YES**
  Replace the existing work station customizing object.

* **NO**
  Do not replace the existing work station customizing object.

**Examples**

**CRTWSCST WSCST(MYLIB/MYWSCOBJ)**

This command creates a work station customizing object named MYWSCOBJ in library MYLIB.

**Error messages**

None
Convert CL Source (CVTCLSRC)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Convert CL Source (CVTCLSRC) command is used to convert Control Language (CL) source code from System/38 syntax to the syntax used on the iSeries system. The following are converted:

- **System/38 object-name.library name** to
  - library-name/object-name
- **System/38 job-name.user-name.job-number** to
  - job-number/user-name/job-name
- Starting comment delimiters: /* to */
- Command names
- Keyword names and values
- Missing required parameters for iSeries commands may need to be added.

Conversion of user-defined commands is limited to the reordering of qualified names and adjusting comment syntax.

The CVTCLSRC command creates a report indicating the success or failure of the source file conversion. This report is contained in a printer file with the name ‘CVTCLSRC’. Successful conversions of System/38 source are noted in the report with the message:

CPF0786 Member has been converted.

Error messages are printed for unsuccessful conversions. Some examples of functions which cannot be converted and may be printed as error messages in the report are:

- stmt# CPF0785 Command cannot be converted
- stmt# CPF0789 Keyword cannot be converted

The user may write a program, perhaps by using the Copy Spooled File (CPYSPLF) command, to process the report based on the success or failure of the conversion.

**Restrictions:** Library QSYS38 must exist on the system to support the conversions and to detect unsupported functions. Commands with unsupported command name, keyword names, or keyword values are not converted.

### Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>FROMFILE</strong></td>
<td>From file</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: From file</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
</tr>
<tr>
<td><strong>TOFILE</strong></td>
<td>To file</td>
<td>Qualified object name</td>
<td>Required, Positional 2</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: To file</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
</tr>
</tbody>
</table>
### FROMMBR

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>FROMMBR</td>
<td>From member</td>
<td>Single values: *ALL Other values (up to 50 repetitions): Generic name, name</td>
<td>Required, Positional 3</td>
</tr>
</tbody>
</table>

#### From file (FROMFILE)

Specifies the System/38 CL source file to have its syntax converted.

This is a required parameter.

**Qualifier 2: Library**

- ***LIBL**  All libraries in the library list for the current thread are searched until the first match is found.
- ***CURLIB** The current library for the thread is used to locate the source file. If no library is specified as the current library for the thread, the QGPL library is used.

**name** Specify the name of the library to be searched.

#### To file (TOFILE)

Specifies the file to contain the converted CL source. It must be different than the name specified for the FROMFILE parameter.

This is a required parameter.

**Qualifier 2: Library**

- ***LIBL**  All libraries in the library list for the current thread are searched until the first match is found.
- ***CURLIB** The current library for the thread is used to locate the source file. If no library is specified as the current library for the thread, the QGPL library is used.

**name** Specify the name of the library to be searched.

#### Member (FROMMBR)

Specifies the members of the file specified for the **From file (FROMFILE)** parameter that are to be converted.

- ***ALL**  All members of the specified source file are to be converted.

**generic-name**  Specify the generic name of the members to be converted.

**name** Specify the names of the members to be converted. Specify no more than fifty names. Note that the member name of the converted source member is the same as the member name of the unconverted source member in the file specified for the FROMFILE parameter.
Examples

CVTCLSRC FROMFILE(OLDLIB/FILEA) TOFILE(NEWLIB/FILEB)
FROMMBR(PGM1 PGM2 PGM3)

This command converts three members (PGM1, PGM2, PGM3) of a System/38 source file (FILEA) located in library OLDLIB, to an iSeries source file. The converted source file members are located in FILEB, in library NEWLIB. The converted members keep their original member names, PGM1, PGM2, and PGM3.

Error messages

*ESCAPE Messages

CPF0781
   File &1 in library &2 not a source file.

CPF0784
   Specified to-file same as from-file.
Convert Date (CVTDAT)

Where allowed to run: Compiled CL program or interpreted
REXX (*BPGM *.IPGM *.REXX *.IREXX)
Threadsafe: No

The Convert Date (CVTDAT) command converts the format of a date value from one format to another, without changing its value. The command ignores any date separators used in the old format, but if separators are included in the converted result, a separator character can be specified on the command.

Only valid dates can be converted. If either the from-format or the to-format use only 2 digits to specify the year (for example, *MDY, *DMY, *YMD, or *JUL), valid dates are in the range of January 1, 1940, to December 31, 2039. Otherwise, valid dates are in the range of August 24, 1928, to May 9, 2071. If the year is specified with only 2 digits, years in the range of 40 to 99 are assumed to be 1940 to 1999; years in the range 00 to 39 are assumed to be 2000 to 2039. The command works in conjunction with the QLEAPADJ system value.

Restrictions: This command is valid only within a CL procedure.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>DATE</td>
<td>Date to be converted</td>
<td>Character value</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td>TOVAR</td>
<td>CL var for converted date</td>
<td>Character value</td>
<td>Required, Positional 2</td>
</tr>
<tr>
<td>TOSEP</td>
<td>To date separator</td>
<td>*SYSVAL, *NONE, *JOB, *BLANK, '/', &quot;,&quot;, &quot;,&quot;, &quot;,&quot;</td>
<td>Optional, Positional 5</td>
</tr>
</tbody>
</table>

Date to be converted (DATE)

Specifies the constant or CL variable containing the date to be converted. When a constant is specified that contains separator characters, it must be enclosed in apostrophes (the separator characters are ignored in the conversion). If separators are used in a constant, leading zeros in each part of the date can be omitted (3/3/88 or 03/03/88 are both valid). If a variable is specified, it must be long enough to contain the date type and its date separators, if used. The valid date separators are the slash (/), hyphen (-), period (.), and comma (,). A variable containing all blanks (X'40') is considered to have a date of length zero, and is not valid.

This is a required parameter.
**CL var for converted date (TOVAR)**

Specifies the name of the CL variable that contains the converted date value.

For every format except Julian, the month and day subfields in the converted result are each 2 bytes in length, are right-justified, and (if necessary) a leading zero is used as a padding character to fill each 2-byte field.

For the Julian and long Julian formats, the day field is 3 bytes long and padded with leading zeros (if necessary). The year field is 2 bytes long for Julian and 4 bytes long for long Julian.

Use the following tables to determine the required minimum length of the variable.

### Field Size and Minimum Variable Length

<table>
<thead>
<tr>
<th>TO FMT</th>
<th>TO SEP</th>
<th>Minimum Variable Length</th>
</tr>
</thead>
<tbody>
<tr>
<td>*JUL</td>
<td>*NONE</td>
<td>5</td>
</tr>
<tr>
<td>*JUL</td>
<td>Any</td>
<td>6</td>
</tr>
<tr>
<td>*MDY, *DMY, *YMD</td>
<td>*NONE</td>
<td>6</td>
</tr>
<tr>
<td>*MDY, *DMY, *YMD</td>
<td>Any</td>
<td>8</td>
</tr>
<tr>
<td>*MDYY, *DMYY, *YYMD</td>
<td>*NONE</td>
<td>8</td>
</tr>
<tr>
<td>*MDYY, *DMYY, *YYMD</td>
<td>Any</td>
<td>10</td>
</tr>
<tr>
<td>*CYMD</td>
<td>*NONE</td>
<td>7</td>
</tr>
<tr>
<td>*CYMD</td>
<td>Any</td>
<td>9</td>
</tr>
<tr>
<td>*LONGJUL</td>
<td>*NONE</td>
<td>7</td>
</tr>
<tr>
<td>*LONGJUL</td>
<td>Any</td>
<td>8</td>
</tr>
<tr>
<td>*JOB</td>
<td>Depends on job date format</td>
<td></td>
</tr>
<tr>
<td>*SYSVAL</td>
<td>Depends on value of QDATFMT</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>TO FMT</th>
<th>Month</th>
<th>Day</th>
<th>Year</th>
</tr>
</thead>
<tbody>
<tr>
<td>*JUL</td>
<td>N/A</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>*MDY, *DMY, *YMD</td>
<td>2</td>
<td>2</td>
<td>2</td>
</tr>
<tr>
<td>*MDYY, *DMYY, *YYMD</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>*ISO, *USA, *EUR, *JIS</td>
<td>2</td>
<td>2</td>
<td>4</td>
</tr>
<tr>
<td>*CYMD</td>
<td>2</td>
<td>2</td>
<td>2 (+1 byte century field)</td>
</tr>
</tbody>
</table>

This is a required parameter.

---

**From date format (FROMFMT)**

Specifies the current format of the date to be converted.

*JOB  The date has the format specified by the job attribute, DATFMT.

*SYSVAL  The date has the format specified by the system value, QDATFMT.

*MDY  The date has the month, day, year format, mmddyy.

*MDYY  The date has the month, day, year format, mmddyyyy.

*DMY  The date has the day, month, year format, ddmmyy.
To date format (TOFMT)

Specifies the format to which the date is to be converted.

*JOB  The date format is converted to the format specified by the job attribute, DATFMT.
*SYSVAL  The date format is converted to the format specified by the system value, QDATFMT.
*MDY  The date format is converted to the month, day, year format, mmddyy.
*MDYY  The date format is converted to the month, day, year format, mmddyyyy.
*DMY  The date format is converted to the day, month, year format, ddmmyy.
*DMYY  The date format is converted to the day, month, year format, ddmmyyyy.
*YMD  The date format is converted to the year, month, day format, yymmd.
*YYMD  The date format is converted to the year, month, day format, yyyyymmd.
*CYMD  The date format is converted to the century, year, month, day format, cyymmd, where c is 0 for years 1928 through 1999 and is 1 for years 2000 through 2071. If the year in the current format is only 2 digits, c will be set to 0 for years 40 through 99 and to 1 for years 00 through 39.
*JUL  The date format is converted to the Julian format, yyddd.
*ISO  The date format is converted to the International Organization for Standardization (ISO) date format, yyyy-mm-dd.
*USA  The date format is converted to the United States date format, mm/dd/yyyy.
*EUR  The date format is converted to the European date format, dd.mm.yyyy.
*JIS  The date format is converted to the Japanese Industrial Standard date format, yyyy-mm-dd.
*LONGJUL
The date has the long Julian format, yyyyddd.

**To date separator (TOSEP)**

Specifies the type of date separators (if any) used in the converted date.

*JOB The converted date has the separators specified by the job attribute, DATSEP.

*SYSVAL The converted date has the separators specified by the system value, QDATSEP.

*NONE No separator characters are contained in the converted date.

*BLANK A blank space is used as the date separator in the converted date.

**separator-character**

Specify the character that is used as the date separator in the converted date. The valid separator characters are the slash (/), hyphen (-), period (.), and comma (,).

**Examples**

Example 1: Converting to DMY Format

```plaintext
DCL VAR(&DATE) TYPE(*CHAR) LEN(8)
:
CVTDAT DATE('12-24-88') TOVAR(&DATE) TOFMT(*DMY)
```

This command converts the date 12-24-88, which is in the MDY format. Because the FROMFMT parameter was not specified, its default, *JOB, indicates that the job attribute DATFMT contains the MDY format. The date is converted to the DMY format, and the separator character specified in the job attribute DATSEP is inserted. If DATSEP contains a slash, the converted result is 24/12/88.

Example 2: Converting to Format Specified by Job Attribute

```plaintext
DCL &PAYDAY *CHAR 6
DCL &NEWPDAY *CHAR 6
:
CVTDAT DATE(&PAYDAY) TOVAR(&NEWPDAY)
FROMFMT(*YMD) TOSEP(*NONE)
```

This command converts the format of the date stored in &PAYDAY from year, month, day to the format specified by the job attribute DATFMT. If, for example, DATFMT contains the MDY format, the format of the converted date is month, day, and year. The converted date is stored in the variable &NEWPDAY. Because &NEWPDAY was declared as a 6-character variable, TOSEP(*NONE) is required; the converted result cannot include separator characters.

Example 3: Converting to CYMD format

```plaintext
DCL &NEWDAY1 *CHAR 7
DCL &NEWDAY2 *CHAR 7
:
CVTDAT DATE('01/24/1939') TOVAR(&NEWDAY1)
FROMFMT(*MDY) TOFMT(*CYMD) TOSEP(*NONE)
CVTDAT DATE('01/24/39') TOVAR(&NEWDAY2)
FROMFMT(*MDY) TOFMT(*CYMD) TOSEP(*NONE)
```

These commands convert dates to the CYMD format.
The first CVTDAT command converts the date specified on the DATE parameter from the month, day, 4-digit year format to the century, year, month, day format. Because the year was specified with 4 digits and the first 2 digits are “19”, the century digit is set to “0”, so the value of "NEWDAY1" is “0390124”.

The second CVTDAT command converts the date specified on the DATE parameter from the month, day, year format to the century, year, month, day format. Because the year was specified with only 2 digits and the year is less than 40, the century digit is set to “1”, so the value of "NEWDAY2" is “1390124”.

Error messages

*ESCAPE Messages

CPF0550
Date too short for specified format.

CPF0551
Separators in date are not valid.

CPF0552
Date contains misplaced or extra separators.

CPF0553
Date contains too many or too few numeric characters.

CPF0554
Variable specified too short for converted date format.

CPF0555
Date not in specified format or date not valid.

CPF0556
Date contains two or more kinds of separators.

CPF0557
Date outside allowed range.
Convert Directory (CVTDIR)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Convert Directory (CVTDIR) command works with the conversion of integrated file system directories from the *TYPE1 format to the *TYPE2 format. Directories in the "root" (/), QOpenSys, and user-defined file systems support *TYPE2 directories. *TYPE2 directories are optimized for performance, size, and reliability as compared to directories having the *TYPE1 format.

The CVTDIR command can provide the current directory format of the file systems, or can change the priority of the convert directory function.

Restrictions:
1. The user must have all object (*ALLOBJ) special authority to use this command.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>OPTION</td>
<td>Option</td>
<td>*CHECK, *CHGPTY</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td>RUNPTY</td>
<td>Run priority</td>
<td>1-99, *SAME, *DFT</td>
<td>Optional</td>
</tr>
<tr>
<td>FORMAT</td>
<td>Format</td>
<td>*TYPE2, *TYPE1</td>
<td>Optional</td>
</tr>
<tr>
<td>DETAIL</td>
<td>Detail</td>
<td>*BASIC, *EXTENDED</td>
<td>Optional</td>
</tr>
<tr>
<td>ASP</td>
<td>Auxiliary storage pool ID</td>
<td>1-32</td>
<td>Optional</td>
</tr>
</tbody>
</table>

Option (OPTION)

 Specifies the function to be performed.

*CHECK

The file systems which are currently on the system are checked to determine their current directory format. Message CPIA084 is sent for the "root" (/), and QOpenSys file systems, and for all active auxiliary storage pools on the system identifying their current directory format.

*CHGPTY

Change the run priority of the convert directory function.
Run priority (RUNPTY)

Specifies the new run priority for the convert directory function. This must be specified if OPTION(*CHGPTY) is used.

Note: There may be a delay before the new priority takes effect for the convert directory function.

*SAME
The run priority does not change.

*DFT
The run priority will be reset to the system default.

1-99
Specify the run priority for the convert directory function.

File system (FILESYS)

This parameter is no longer supported and will be ignored. It has been kept strictly for syntactic compatibility with releases prior to Version 5 Release 4 Modification 0 of the i5/OS.

Format (FORMAT)

This parameter is no longer supported and will be ignored. It has been kept strictly for syntactic compatibility with releases prior to Version 5 Release 4 Modification 0 of the i5/OS.

Detail (DETAIL)

This parameter is no longer supported and will be ignored. It has been kept strictly for syntactic compatibility with releases prior to Version 5 Release 4 Modification 0 of the i5/OS.

Auxiliary storage pool ID (ASP)

This parameter is no longer supported and will be ignored. It has been kept strictly for syntactic compatibility with releases prior to Version 5 Release 4 Modification 0 of the i5/OS.

Examples

Example 1: Checking Directory Format Information

CVTDIR OPTION(*CHECK)

This command displays the current directory formats for the file systems.

Example 2: Changing the Run Priority of the Convert Directory Function

CVTDIR OPTION(*CHGPTY) RUNPTY(50)

This command changes the run priority of the convert directory function.
Error messages

*ESCAPE Messages

CPF9890
Function not supported, request rejected.

CPFA099
The requested convert directory option cannot be performed.

CPFA09A
Errors occurred during directory conversion.
Convert DLS Name (CVTDLSNAM)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Convert Document Library Services Name (CVTDLSNAM) command is used before or after a change in the CCSID assumed for EBCDIC object names by QDLS (the document library services file system). The command can help identify QDLS object names that may be different because of the CCSID change, and it can rename QDLS objects so they retain their original names. It can also correct the names of objects whose names changed when they were copied between QDLS and another file system. The command prints a report with the job’s spooled output that lists selected objects and any actions taken on those objects.

When converting object names to or from EBCDIC, QDLS uses the job default CCSID unless data area QUSRYS/QODEC500 exists, in which case QDLS uses CCSID 500 (the data area allows reversion to the behavior of early versions of QDLS). The CCSID used by QDLS is therefore changed by creating or deleting the data area, or by changing the job default CCSID when the data area does not exist.

The CCSID affects the view of QDLS object names by integrated file system clients of QDLS, which must convert object names to and from EBCDIC. Those clients include:
- Integrated file system commands such as DSPLNK, CPY, MOV, and RNM
- UNIX-type APIs provided by the integrated file system, such as access, open, rename, and unlink
- IBM eServer iSeries Access for Windows

The CCSID does not affect clients of QDLS that work directly with EBCDIC object names, which include:
- Document and folder commands, such as CRTDOC, CPYDOC, WRKDOC, CRTFLR, WRKFLR, DLTDLO, and RNMDLO
- Hierarchical file system (HFS) APIs, such as QHFDLTSF, QHFOPNDR, QHFOPNSE, and QHFRNMSF

Even for integrated file system clients of QDLS, the CCSID doesn’t matter except for objects that are also used by EBCDIC clients. In that case, QDLS object names may appear different to the clients if the names contain variant characters and the clients are using different CCSIDs (integrated file system clients use the CCSID as described earlier, and EBCDIC clients likely use the job default CCSID).

Restrictions:
- You must have read (*R) authority to the directory containing the object links and execute (*X) to the other directories in the path.
- The additional authority restrictions from the RNM command apply when renaming objects.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>OBJ</td>
<td>Object</td>
<td>Values (up to 300 repetitions): Path name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td>SUBTREE</td>
<td>Directory subtree</td>
<td>*OBJ, *DIR, *ALL</td>
<td>Optional</td>
</tr>
<tr>
<td>ACTION</td>
<td>Action</td>
<td>*LIST, *RENAME</td>
<td>Optional</td>
</tr>
<tr>
<td>PREVIEW</td>
<td>Preview results</td>
<td>*NO, *YES</td>
<td>Optional</td>
</tr>
</tbody>
</table>
### Object (OBJ)

Specifies the objects to process. A maximum of 300 path names can be specified; however, all paths must be for the same file system. Each path name can be either a simple name or a name that is qualified with the name of the directory in which the object is located. A pattern can be specified in the last part of the path name: an asterisk (*) matches any number of characters and a question mark (?) matches any single character. If a path name is qualified or contains a pattern, it must be enclosed in apostrophes (’).

For more information on specifying path names, refer to "Object naming rules" in the CL concepts and reference topic in the iSeries Information Center at http://www.ibm.com/eserver/iseries/infocenter.

The CVTDLSNAM command is an integrated file system client of QDLS. As such, its view of object names can be different than that of EBCDIC clients. So, use care in specifying object names. It is generally safer to use generic characters in place of variant characters (for example, specify X?X as an object name rather than XIX).

### Directory subtree (SUBTREE)

Specifies whether directory subtrees are processed.

- **OBJ** Only the objects that match the given path names are processed. If a path name specifies a directory, objects in the directory are not processed.
- **DIR** Objects in the first level of each directory that matches a given path name are processed.
- **ALL** The entire subtree of each directory that matches a given path name is processed.

### Action (ACTION)

Specifies the action to perform on the selected objects.

- **LIST** For QDLS, this value lists the selected objects that might appear to have different names if the CCSID assumed by QDLS for EBCDIC object names is changed from the specified old value to the specified new value. For other file systems, this value lists the selected objects that might have an unexpected name after having been copied from QDLS, and neither specified CCSID is used in this case.

- **RENAME** Corrects the names of the selected objects. If *RENAME* is used more than once on an object, the results will probably not be meaningful.

  Some objects may fail to be renamed when requested, such as if the new name already exists. However, the command will not fail immediately; it will continue to process any remaining objects.
For QDLS, *RENAME will change the names such that, after changing the CCSID assumed by QDLS for EBCDIC object names from the specified old value to the specified new value, the object names will appear the same as before the change to integrated file system clients of QDLS.

For other file systems, the specified objects are presumed to have been created with the specified old CCSID and implicitly renamed as they were copied from QDLS by an integrated file system client of QDLS using the specified new CCSID. *RENAME will change the names of the objects to be the same as those of the original QDLS objects.

**Note:** The effect of a rename can be undone by another rename with the CCSIDs reversed. For example, if a rename is done using FROMCCSID(500) and TOCCSID(273), the original name(s) can be restored by a rename using FROMCCSID(273) and TOCCSID(500).

---

**Preview results (PREVIEW)**

Selects whether to preview the results of the selected action.

*NO Perform the selected action.

*YES Inhibit the selected action and report what the results would be. This value is allowed only when ACTION(*RENAME) is specified.

---

**From CCSID (FROMCCSID)**

Specifies the original coded character set identifier (CCSID) of the EBCDIC object name. This value is ignored when processing objects in file systems other than QDLS if ACTION is *LIST.

500 CCSID 500 is used. That is the CCSID used by early versions of QDLS.

*JOB The current job’s default CCSID is used.

*SYSVAL The CCSID specified in the system value QCCSID is used.

*HEX or 65535 The CCSID currently assumed by QDLS for EBCDIC object names is used.

1-65535 Specify the CCSID to be used. More information on valid CCSIDs is in the Globalization topic in the iSeries Information Center at http://www.ibm.com/eserver/iseries/infocenter.

---

**To CCSID (TOCCSID)**

Specifies the coded character set identifier (CCSID) assumed by QDLS for EBCDIC object names. This value is ignored when processing objects in file systems other than QDLS if ACTION is *LIST.

*JOB The current job’s default CCSID is used.

*SYSVAL The CCSID specified in the system value QCCSID is used.

*HEX or 65535 The new CCSID is unknown, such as when different jobs will be using different CCSIDS. This value may not be used with ACTION(*RENAME).
Examples

Example 1: List QDLS Objects Affected by a CCSID Change

A new release of the operating system is installed and QDLS now assumes the job default CCSID instead of CCSID 500 for EBCDIC object names. The job CCSID is currently set to 37. The following command is used to identify the objects that effectively have new names for integrated file system clients of QDLS. Note that the ACTION, FROMCCSID, and TOCCSID parameters could all have been omitted from the command, since they specify the default values in this case.

```cvtldsnam  obj('/qdl') subTree(*All) action(*List)
             fromccsid(500) toccsid(37)
```

Output similar to this might be produced:

```
/QDLS/FLRA/X! --> X!
/QDLS/FLRB/X! --> X!
```

Each line shows two names for an object, as it would be seen by clients using CCSID 500 and CCSID 37 (the second name won’t be shown if TOCCSID is *HEX). The output shows that two objects are affected by the change of the assumed CCSID. The object known before the change as XJ by integrated file system clients is known as X! afterward, and X! is renamed to XJ.

The name XJ seems more reasonable than either X| or X!, so we assume X! is the correct name in both cases. In the first case the new name is desirable; we surmise the object was created as X| by a client using CCSID 37. In the second case the new name is undesirable; the object was presumably created by a client using CCSID 500.

Example 2: Rename QDLS Object to Adjust for a CCSID Change

The second object name from the example above is corrected using the following command. For this example the job CCSID is 500 (necessary to guarantee correct recognition of the object name XJ). It is likely that a generic name (such as * or X? instead of X!) would be used in similar situations, eliminating the need to adjust the job CCSID.

```cvtldsnam  obj('/QDLS/FLRB/X!') action(*rename)
             fromccsid(500) toccsid(37)
```

This output might be produced:

```
/QDLS/FLRB/X! --> X
```

Each line again shows two names for an object, but this time both names are what a CCSID 500 client would see. The output indicates that XJ is renamed to X|. The new name may appear incorrect to a CCSID 500 client (XJ), but it will appear as desired to a CCSID 37 client (X!).

Error messages

*STATUS Messages
CPI8A22
Processing &1.

*ESCAPE Messages
CPF8AC0
&1 command failed.
Convert Education (CVTEDU)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Convert Education (CVTEDU) command converts the online education courses that are stored in a library into a format that can be used on the AS/400 system. This command converts course modules from ASCII to EBCDIC.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>COURSE</td>
<td>Course ID</td>
<td>Name, *ALLADDED</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td>LNG</td>
<td>Language ID</td>
<td>*SYSVAL, 2922, 2923, 2924, 2925, 2926, 2928, 2929, 2931, 2932, 2933, 2937, 2938, 2939, 2940, 2942, 2950, 2956, 2957, 2958, 2962, 2963, 2961, 2966, 2980, 2981, 2984, 2986, 2987, 2989, 2996</td>
<td>Optional</td>
</tr>
</tbody>
</table>

Course ID (COURSE)

Specifies which course you want to convert.

The possible values are:

*course-ID*

Specify a specific course you want to convert using the name of the library where the course modules are stored.

*ALLADDED*

All courses previously added to the system are converted.

Language ID (LNG)

Specifies the language ID that you want to use to convert the courses.

The possible values are:

*SYSVAL*

The current primary language setting for the system is used.

*language-ID*

Specify a language ID for the system. This language ID is a 4-digit number assigned to each specific language. To view the list of languages and their identifying numbers, move the cursor to the Language ID parameter field and press the F4 (Prompt) key when you are on the command prompt display.
Examples

CVTEDU COURSE(*ALLADDED)

This command converts all of the courses previously added through the education administration system from ASCII to EBCDIC.

Error messages

*ESCAPE Messages

CPF1D47
Not authorized to use CVTEDU command.

CPF1D49
Errors occurred during command processing.
Convert IP Address (CVTIPSIFC)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Convert IP over SNA Interface (CVTIPSIFC) command converts an IP (Internet Protocol) address into its associated SNA network identifier and location name. The location entries defined with the Add IP over SNA Location Entry (ADDIPSLOC) CL command are searched to find the SNA location name and SNA network identifier associated with the input internet address (INTNETADR).

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>INTNETADR</td>
<td>Internet address</td>
<td>Character value</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td>OUTPUT</td>
<td>Output</td>
<td>* , *PRINT</td>
<td>Optional, Positional 2</td>
</tr>
</tbody>
</table>

Internet address (INTNETADR)

Specifies the internet address of the local host or a remote host to be converted. The internet address is specified in the form nnn.nnn.nnn.nnn, where nnn is a decimal number ranging from 0 through 255. An internet address is not valid if it has a value of all binary ones or all binary zeros for the network identifier (ID) portion or the host ID portion of the address. If the internet address is entered from a command line, the address must be enclosed in apostrophes.

This is a required parameter.

Restrictions:

1. The internet address cannot begin with 0 (for example, 0.nnn.nnn.nnn).
2. The internet address cannot begin with 127 (for example, 127.nnn.nnn.nnn). This address range is reserved for TCP/IP loopback addresses.
3. The internet address cannot be a class D or class E address. Valid class D addresses range from 224.nnn.nnn.nnn to 239.nnn.nnn.nnn. Valid class E addresses range from 240.nnn.nnn.nnn to 255.nnn.nnn.nnn.

Output (OUTPUT)

Specifies where the result should be returned.

The possible values are:

© Copyright IBM Corp. 1998, 2006
* The output is displayed (if requested by an interactive job) or printed with the job’s spooled output (if requested by a batch job).

*PRINT
   The output is printed with the job’s spooled output.

---

**Examples**

**Example 1: Printing a Converted IP Address**

```
CVTIPSIFC  INTNETADR('128.1.2.3') OUTPUT(*PRINT)
```

This command finds the SNA network identifier and location name associated with IP address 128.1.2.3 and puts the result in the job’s spooled output.

**Example 2: Displaying a Converted IP Address**

```
CVTIPSIFC  INTNETADR(128.2.3.4)
```

This command finds the SNA Network Identifier and Location Name associated with IP address 128.2.3.4 and puts the result to the display for an interactive job.

---

**Error messages**

***ESCAPE Messages**

**CPFA111**
   Internet address not converted.

**CPFA118**
   No associated SNA network identifier and location name found.
Convert Network ID / Location (CVTIPSLOC)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Convert IP over SNA Location Entry (CVTIPSLOC) command is used to convert a specified SNA network identifier and location name into one or more associated IP addresses. The location entries defined with the ADD IP over SNA Location Entry (ADDIPSLOC) CL command are searched to find one or more IP addresses that are associated with the input SNA location name (LOC) and SNA network identifier (NETID).

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>NETID</td>
<td>Network identifier</td>
<td>Communications name, *NETATR</td>
<td>Optional, Positional 1</td>
</tr>
<tr>
<td>LOC</td>
<td>Location name</td>
<td>Communications name</td>
<td>Optional, Positional 2</td>
</tr>
<tr>
<td>OUTPUT</td>
<td>Output</td>
<td>*PRINT</td>
<td>Optional, Positional 3</td>
</tr>
</tbody>
</table>

Network identifier (NETID)

Specifies the SNA network identifier for the local host or a remote host.

The possible values are:

*NETATR

The network identifier in the network attributes for this host is used.

network-identifier

Specify the network identifier for the local host or a remote host. The network identifier can be one to eight characters in length. The first character must be A (or a) through Z (or z), or special characters $, #, or @ followed by 0 through 9, A (or a) through Z (or z), $, #, or @.

Location name (LOC)

Specifies the SNA location name to be converted.

This is a required parameter.

The possible values are:

location-name

Specify the SNA location name for the local host or a remote host. This name can be one to eight
characters in length. The first character must be A (or a) through Z (or z), or special characters $, #, or @ followed by 0 through 9, A (or a) through Z (or z), $, #, or @.

Output (OUTPUT)

Specifies where the results are returned.

The possible values are:

* The output is displayed (if requested by an interactive job) or printed with the job’s spooled output (if requested by a batch job).

*PRINT
The output is printed with the job’s spooled output.

Examples

CVTIPSLOC LOC(LUNAMEX) OUTPUT(*PRINT)

This command converts the location name LUNAMEX with the default network identifier specified in the network attributes and places the results in the job’s spooled output.

Error messages

*ESCAPE Messages

CPFA115
SNA network identifier and location name not converted.

CPFA119
No associated internet address found.
Convert Optical Backup (CVTOPTBKU)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Convert Optical Backup (CVTOPTBKU) command converts an optical backup volume to an optical primary volume. User applications and programs can then write to the converted volume.

Note: Once an optical volume is converted from a backup volume to a primary volume, you must initialize the optical volume to convert it to a backup volume again. Initializing an optical volume results in losing all existing information on the optical volume.

Restriction: To use this command you must have *ALL authority to the authorization list securing the volume to be converted.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>BKUVOL</td>
<td>Backup volume identifier</td>
<td>Character value</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td>PRIVOL</td>
<td>Primary volume identifier</td>
<td>Character value, *PRVPRIVOL</td>
<td>Optional, Positional 2</td>
</tr>
</tbody>
</table>

Backup volume identifier (BKUVOL)

Specifies the volume identifier of the optical backup volume being converted to a primary volume.

Primary volume identifier (PRIVOL)

Specifies the identifier of the optical volume after it is converted to a primary volume.

Note: The identifier must be unique within the system you are using. More information about optical volume names can be found in the Optical Support, SC41-4310 book.

*PRVPRIVOL

The identifier of the new primary optical volume is the same as the identifier of the primary optical volume for which this volume previously was a backup.

When an optical backup volume is first used, the system records the volume identifier of the primary volume on the media. This is done to ensure that no other primary volume can use the same backup volume identifier. This also ensures that the original name of the primary volume is known at the time the optical backup volume is converted.
**primary-volume-identifier**

Specify a new volume identifier. This is the identifier of the volume after this command completes successfully.

---

**Examples**

CVTOPTBKU BKUVOL(VOL01BACKUP) PRIVOL(VOL02)

This command converts the optical backup volume VOL01BACKUP to a primary optical volume. VOL02 is the identifier of the optical volume after it is converted.

---

**Error messages**

*ESCAPE Messages*

OPT1305  
Optical volume &1 is read only.

OPT1315  
Optical volume &1 is write protected.

OPT1320  
Optical volume &1 in use.

OPT1325  
Optical volume format not recognized.

OPT1330  
Optical volume not found or not useable.

OPT1331  
Optical volume &1 not found.

OPT1340  
Optical volume &1 not initialized.

OPT1342  
Invalid volume identifier specified.

OPT1345  
No free space available on media.

OPT1350  
Write operation failed to optical volume &1.

OPT1360  
Media directory corrupted on optical volume &1.

OPT1375  
Optical volume &1 already exists.

OPT1460  
Optical volume &1 is not in an optical device.

OPT1462  
Operation not completed, optical volume is not a backup volume.

OPT1530  
&1 does not represent a valid optical device.
OPT1605
Media or device error occurred.

OPT1790
Operation not allowed or conflicts with another request.

OPT1805
Error accessing optical volume index file.

OPT1810
Error accessing optical directory index file.

OPT1815
Internal program error occurred.

OPT1820
Internal error occurred on optical device &1.

OPT1825
Optical indexes are incorrect for optical device &1.

OPT1860
Request to optical device &1 failed.

OPT1861
No device description configured for resource &1.

OPT1862
No active device description for resource &1.

OPT1863
Optical libraries need to be reclaimed.

OPT1872
Optical request timed out or was cancelled.

OPT2030
Error during Convert Optical Backup.

OPT2301
Internal system object in use.

OPT7740
User not authorized to object &2 in library &3 type &4.
The Convert Performance Data (CVTPFRDTA) command converts performance data from a previous release to the format needed for processing by the current release of Performance Tools for iSeries(TM).

The command first determines the release level at which the data was collected. Then the members of all necessary files are converted. The conversion may be done in the same library where the current data resides. To avoid the risk of destroying the old data if the command ends abnormally, convert the data into a different library (TOLIB), and later, delete the data from the old library (FROMLIB).

If the conversion is done in a different library, the old data remains in the current library (FROMLIB) and the new data resides in the new library (TOLIB). If a new library is specified for the newly converted data, all files are copied to the new library, including those files which do not need to be converted.

### Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>FROMLIB</td>
<td>From library</td>
<td>Name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td>TOLIB</td>
<td>To library</td>
<td>Name</td>
<td>Required, Positional 2</td>
</tr>
<tr>
<td>JOBD</td>
<td>Job description</td>
<td>Single values: *USRPRF, *NONE Other values: Qualified object name</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Job description</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
</tr>
</tbody>
</table>

#### From library (FROMLIB)

Specifies the library that contains the files to be converted.

This is a required parameter.

*name* Specify the name of the library where the files to be converted are located.

#### To library (TOLIB)

Specifies the library where the converted files will be located.

This is a required parameter.

*name* Specify the name of the library where the converted files are to be located.
Job description (JOBD)

Specifies the job description used to submit jobs for batch processing.

Single values

*USRPRF
   The job description specified in the user’s user profile is used.

*NONE
   A batch job is not submitted. Instead, processing continues interactively while the user waits. The user’s work station is not available for other use during this time, which can be significant for long jobs.

Qualifier 1: Job description

ame
   Specify the name of the job description to be used.

Qualifier 2: Library

*LIBL
   All libraries in the library list for the current thread are searched until the first match is found.

*CURLIB
   The current library for the job is used to locate the job description. If no current library entry exists in the library list, the QGPL library is used.

name
   Specify the name of the library to be searched.

Examples

Example 1: Converting Data Using a Batch Job
CVTPFRDTA FROMLIB(MIKE) TOLIB(TERESA)

This command converts the performance data in library MIKE and places it in library TERESA. The conversion is done by a batch job submitted using the job description associated with the user profile of the user running this command.

Example 2: Converting Performance Data Interactively
CVTPFRDTA FROMLIB(QPFRDATA) TOLIB(QPFRDATA) JOBD(*NONE)

This command converts the performance data in library QPFRDATA and places it in the same library after conversion is complete. This conversion occurs interactively while the user waits.

Error messages

*ESCAPE Messages

CPF0A0B
   Performance tools files did not convert.

CPF22F7
   Number of authorities must be between 1 and &1.
CPF22FA
Authority value &1 not valid.

CPF22FB
Must specify *EXCLUDE or *AUTL as only authority value.

CPF2817
Copy command ended because of error.

CPF4102
File &2 in library &3 with member &4 not found.

CPF8122
&8 damage on library &4.

CPF9801
Object &2 in library &3 not found.

CPF9802
Not authorized to object &2 in &3.

CPF9803
Cannot allocate object &2 in library &3.

CPF9807
One or more libraries in library list deleted.

CPF9808
Cannot allocate one or more libraries on library list.

CPF9810
Library &1 not found.

CPF9811
Program &1 in library &2 not found.

CPF9812
File &1 in library &2 not found.

CPF9820
Not authorized to use library &1.

CPF9830
Cannot assign library &1.
Convert Pfr Thread Data (CVTPFRTHD)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Convert Performance Thread (CVTPFRTHD) command converts job performance data records. It supports data collected by the STRPFRMON (Start Performance Monitor) command or data generated by the CRTPFRDTA (Create Performance Data) command.

The specified member (MBR parameter) of file QAPMJOBS or QAPMJOBL contains records with thread-level performance data. You can use CVTPFRTHD to convert this data and write the resulting records to a member by the same name (MBR parameter) in file QAPMTJOB. The output file member will contain records with job-level performance data which are a total of the performance information for all threads running within the job.

The input file (QAPMJOBS or QAPMJOBL) must exist in the library specified on the LIB parameter. If file QAPMTJOB does not exist in the specified library (LIB parameter), it will be created automatically. A file member by the name specified (MBR parameter) will be automatically added to file QAPMTJOB if it did not already exist.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>MBR</td>
<td>Member</td>
<td>Name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td>LIB</td>
<td>Library</td>
<td>Name, QFRDATA</td>
<td>Optional</td>
</tr>
<tr>
<td>REPLACE</td>
<td>Replace</td>
<td>*YES, *NO</td>
<td>Optional</td>
</tr>
</tbody>
</table>

Member (MBR)

Specifies the member of file QAPMJOBS or file QAPMJOBL that contains the collections to be processed. This member will be created, if it does not already exist, or replaced in QAPMTJOB file.

*name* Specify the name of the member containing thread-level performance data.

Library (LIB)

Specifies the library where the input file resides, and where the QAPMTJOB file either resides or will be created.

QFRDATA  IBM-supplied performance data library QFRDATA is to be used to locate the input database file.

*name* Specify the name of the library where the input database file is located.
**Replace (REPLACE)**

Specifies whether the specified member in file QAPMTJOB will be replaced.

*YES*  If the member did not exist before, it is created. If the member already exists, the data contained in it is replaced.

*NO*  If the member did not exist before, it is created. If the member already exists, the data contained in it is not replaced and an error message is signalled.

---

**Examples**

CVTPFRTHD MBR(MYDATA)

This command converts performance data records. Member MYDATA in file QAPMJOBL or file QAPMJOBS in library QFFRDATA contains the collections to be processed.

---

**Error messages**

*ESCAPE Messages*

CPF0A83  Performance thread data not converted.

CPF0A84  Member already exists.

CPF0A85  User profile &1 is not authorized to library &2.

CPF2110  Library &1 not found.

CPF2817  Copy command ended because of error.

CPF5030  Partial damage on member &4.

CPF9810  Library &1 not found.

CPF9812  File &1 in library &2 not found.

CPF9845  Error occurred while opening file &1.

CPF9846  Error while processing file &1 in library &2.
Convert RPC Source (CVTRPCSRC)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Convert RPC Source (CVTRPCSRC) command generates C code from an input file written in the Remote Procedure Call (RPC) Language. The generated C code can be used to implement an RPC protocol.

This command is equivalent to running the **rpcgen** utility on a UNIX system.

This command can also be issued using the following alternative command name:
• **RPCGEN**

**Restrictions:**
1. The user must have execute (*X) authority to each directory in the path for both the input and output files.
2. The user must have read (*R) authority to the input file.
3. The user must have write, execute (*WX) authority to the output file directory.

**Parameters**

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>FROMFILE</td>
<td>From file</td>
<td><em>Path name</em></td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td>PROTOCOL</td>
<td>Protocol</td>
<td>Values (up to 2 repetitions): *NONE, *TCP, *UDP</td>
<td>Optional</td>
</tr>
<tr>
<td>TOFILE</td>
<td>To file</td>
<td><em>Path name</em></td>
<td>Optional</td>
</tr>
</tbody>
</table>

**From file (FROMFILE)**

Specifies the path name of the input source file written in the Remote Procedure Call (RPC) Language. The input source file must be a file in the "root" (/) or QOpenSys file system.


**Option (OPTION)**

Specifies the compile options.
*NOSAMP
   All file types except the sample files (*CLTSAMP and *SVRSAMP) are generated.

*ALL       All file types are generated.

*XDR       The input file is compiled into XDR (eXternal Data Representation) routines.

*HDR       The input file is compiled into C data-definitions (a header file).

*CLTSTUB   The input file is compiled into client-side stub procedures.

*SVRSTUB   The input file is compiled into server-side stub procedures. However, no "main" routine is generated.

*CLTSAMP   Sample client code that uses remote procedure calls is generated. The file can be customized for the application.

*SVRSAMP   Sample server code that uses remote procedure calls is generated. The file can be customized for the application.

---

**Protocol (PROTOCOL)**

Compiles into server-side stub procedures for the transport that is specified. The specified value must be present in the /etc/netconfig file at the time the server application is run. This parameter is only valid when OPTION(*SVRSTUB) is specified. One or more of the following options may be specified:

*NONE      Compile server-side stub procedures for all transports that are in the /etc/netconfig file.

*TCP       Compile server-side stub procedures for the TCP transport.

*UDP       Compile server-side stubs for the UDP transport.

---

**To file (TOFILE)**

Specifies the path name of the output file. This option is only allowed if OPTION(*ALL) or OPTION(*NOSAMP) is not specified. When OPTION(*ALL) or OPTION(*NOSAMP) is specified, or if the TOFILE parameter is not specified when using another option, the From file (FROMFILE) parameter is used to generate the TOFILE name as follows, where filename is the name of the input file name from the FROMFILE parameter.

- filename.h for a header file
- filename_xdr.c for an XDR file
- filename_clnt.c for client-side stubs
- filename_svc.c for server-side stubs
- filename_client.c for client-side sample files
- filename_server.c for server-side sample files

The output file or files for sample code must not exist; if any of the sample output files exist, the command will fail. Other output files will be overwritten if they exist.
'to-file-path name'
    Specify a path name to be used to generate the TOFILE name or names.
    For more information on specifying path names, refer to "Object naming rules" in the CL concepts and reference topic in the iSeries Information Center at http://www.ibm.com/eserver/iseries/infocenter.

Examples

**Example 1: Convert RPC Source to Default Files**
CVTRPCSRC FROMFILE('/myrpc') OPTION(*ALL)

This converts the RPC language file `/myrpc` into all four file types, *XDR, *HDR, *CLTSTUB and *SVRSTUB. The default PROTOCOL(*TCP) is used to generate the server-side stub programs. The files are placed into the following file names:
- myrpc.h for a header file
- myrpc_xdr.c for an XDR file
- myrpc_clnt.c for client-side stubs
- myrpc_svc.c for server-side stubs

**Example 2: Convert RPC Source to Client Stubs Only**
CVTRPCSRC FROMFILE('/myrpc2') OPTION(*CLTSTUB)
    TOFILE('/myclnt.c')

This converts the RPC language file `/myrpc2` into client-side stub procedures. The results are placed into the file `/myclnt.c` as specified.

Error messages

**ESCAPE Messages**

**CPFB416**
CVTRPCSRC or RPCGEN command failed.
Convert TCP/IP CL Source (CVTTCPCL)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Convert TCP/IP CL Source (CVTTCPCL) command is used to convert TCP/IP commands from releases prior to Version 3, Release 1, Modification 0 (V3R1M0) to the command syntax for the current release. The pre-V3R1M0 TCP/IP CL commands to be converted must exist in a source physical file.

The following commands are converted based on their specified parameter values. In some cases you may need to manually update the commands after conversion. Messages are issued to help identify the command statements that require manual updates.

- ADDTCPLNK
- CHGTCPLNK
- RMVTCPLNK
- STRTCPLNK
- ENDTCPCLNK
- ADDTCPRTE
- CHGTCPRTE
- RMVTCPRTE
- ADDTCPPTPORT
- RMVTCPPTPORT
- ADDTCPRTSI
- RMVTCPRSI
- CHGTCPA
- ENDTCPCNN
- STRTCPTELN
- CHGVT1MAP
- SETVT1MAP
- DSPVT1MAP
- ENDSBS SBS(QTCP)
- STRSBS SBSD(QTCP/QTCP)

The CVTTCPCL command creates a printer file with the name CVTTCPCL. This printer file contains a report that indicates the success or failure of the source file conversion.

If a printer device file with the name CVTTCPCL is found in the job's library list when the CVTTCPCL command is issued, that printer device file is used to create the printer file. Otherwise, the CVTTCPCL command uses the Override with Printer File (OVRPRTF) command to use printer device file QSYS/QSYSPRT to create the printer file.

Note: Use the Create Printer File (CRTPRTF) command to create a printer device file.

Successful conversions of TCP/IP command source are noted in the report with the message:
TCP1E08 Member has been converted.
Informational messages are printed for unsuccessful command conversions. (Informational messages are also sent to the job log during conversion, and a single escape message is sent when the CVTTCPCL command has completed if any informational messages have been sent.) Some examples of functions that cannot be converted and may be printed as informational messages in the report are:

TCP1E07 Command &1 cannot be converted
TCP1E10 Parameter keyword cannot be converted in command &1

The user can write a program, perhaps by using the Copy Spoooled File (CPYSPLF) command, to process the report based on the success or failure of the conversion.

### Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>FROMFILE</td>
<td>From file</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: From file</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>TOFILE</td>
<td>To file</td>
<td>Qualified object name</td>
<td>Required, Positional 2</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: To file</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>FROMMBR</td>
<td>From member</td>
<td>Single values: *ALL Other values (up to 50 repetitions): Generic name, name</td>
<td>Required, Positional 3</td>
</tr>
</tbody>
</table>

### From file (FROMFILE)

Specifies the CL source file containing TCP/IP commands to be converted.

**Qualifier 1: From file**

*name* Specify the name of the CL source file to convert.

**Qualifier 2: Library**

*LIBL* All libraries in the job’s library list are searched until the first match is found.

*CURLIB* The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

*name* Specify the name of the library to be searched.

### To file (TOFILE)

Specifies the file in which the converted source is placed. It must be different than the name of the FROMFILE parameter.

**Qualifier 1: To file**

*name* Specify the name in which the converted source file is placed.
Qualifier 2: Library

*LIBL  All libraries in the job’s library list are searched until the first match is found.

*CURLIB  The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

name  Specify the name of the library to be searched.

From member (FROMMBR)

Specifies the member of the source file member to convert.

*ALL  All members of the specified source file are converted to V3R1M0 TCP/IP command syntax if possible.

generic-name  Specify the generic name of the source file members to convert. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name. If the complete object name is specified and multiple libraries are searched, multiple objects can be converted only if *ALL or *ALLUSR library values can be specified for the name. Only the first 50 members matching the generic name’s criteria will be converted.

name  Specify the names of the source file members to convert. Specify no more than 50 names.

Note that the only source members that are processed are those with a member type of CL, CLP, or TXT. Members in the FROMFILE with any other value for the member type are ignored by the CVTTCPCL command. If a source member is processed, the name of the converted source member in the TOFILE will be the same as the member name in the FROMFILE.

Examples

CVTTCPCL  FROMFILE(OLDLIB/QCLSRC)  TOFILE(NEWLIB/QCLSRC)
FROMMBR(TCPPGM1 TCPPGM2 TCPPGM3)

This command converts all TCP/IP commands in the three members (TCPPGM1, TCPPGM2, TCPPGM3) of a CL source file (QCLSRC) located in library OLDLIB, to their new command names and formats. The converted source file members are located in QCLSRC, in library NEWLIB. The converted members keep their original member names, TCPPGM1, TCPPGM2, and TCPPGM3.

Error messages

*ESCAPE Messages

CPF9801

Object &2 in library &3 not found.

CPF9810

Library &1 not found.
TCP1E02
File &1 in library &2 not found.

TCP1E03
File &1 in library &2 not a source file.

TCP1E06
Specified TOFILE same as FROMFILE.
Convert To Folder (CVTTOFLR)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Convert To Folder (CVTTOFLR) command converts a virtual disk into a folder and PC documents. The files and directories on the virtual disk are copied into the specified folder, which becomes the root directory.

**Note:** Do not precede an entry with an asterisk unless that entry is a "special value" that is shown (on the display itself or in the help information) with an asterisk.

### Error messages for CVTTOFLR

#### *ESCAPE Messages

**IWS1613**
Virtual disk converted. Errors occurred.

**IWS1614**
Unable to convert virtual disk to folder.

#### *STATUS Messages

**IWS1621**
Converting virtual disk to folder.

### Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>FROMVDSK</td>
<td>From virtual disk</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: From virtual disk</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
</tr>
<tr>
<td>TOFLR</td>
<td>To folder</td>
<td>Character value</td>
<td>Required, Positional 2</td>
</tr>
<tr>
<td>REPLACE</td>
<td>Replace documents</td>
<td>*NO, *YES</td>
<td>Optional, Positional 3</td>
</tr>
</tbody>
</table>

**From virtual disk (FROMVDSK)**

Specifies the name and library of the virtual disk that is converted.

This is a required parameter.

The possible library values are:

*LIBL  The library list is used to locate the virtual disk.
*CURLIB
The current library for the job is used to locate the virtual disk. If no current library entry exists in the library list, QGPL is used.

library-name
Specify the library where the virtual disk is located.

To folder (TOFLR)
Specifies the name of the folder that the virtual disk is converted to. This can be a fully qualified path name.

All the files and directories in the virtual disk are converted. All folders except the last folder in the path must already exist. If the last folder does not exist, it is created.

This is a required parameter.

Replace documents (REPLACE)
Specifies if an existing document should be replaced when a file on the virtual disk has the same name as the document.

*NO It is not replaced.
*YES The existing document is replaced by the file from the virtual disk.

Warning:
All documents that have the same name as files on the virtual disk will be replaced without any error messages being sent.

Examples
None

Error messages

*ESCAPE Messages
IWS1613
Virtual disk converted. Errors occurred.
IWS1614
Unable to convert virtual disk to folder.

*STATUS Messages
IWS1621
Converting virtual disk to folder.
Convert User Certificate (CVTUSRCERT)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Convert User Certificate (CVTUSRCERT) command allows converting of user certificates from being stored and mapped locally on the system to using Enterprise Identity Mapping (EIM) for mapping and Lightweight Directory Access Protocol (LDAP) for storage.

Restrictions:
- You must have all object (*ALLOBJ) and security administrator (*SECADM) special authorities.
- Prior to running this command, the digital ID configuration must have been set using the Set Digital ID Configuration Information (QsySetDigitalIDConfig) API or Digital Certificate Manager (DCM).
- You must have configured this system to participate in an EIM domain.
- You must have set the system connection information using the Set EIM Connect Information (QsySetEIMConnectInfo) API or the EIM configuration GUI.
- For the user profile specified on the command, there must be a target association to an EIM identifier for the local registry that was specified when the system was configured to participate in an EIM domain.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>USRPRF</td>
<td>User profile</td>
<td>Simple name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td>OPTION</td>
<td>Option</td>
<td>*CVTRMV, *RMV, *CVT</td>
<td>Optional, Positional 2</td>
</tr>
</tbody>
</table>

User profile (USRPRF)

Specifies the user profile whose certificates are to be converted.

This is a required parameter.

simple-name
   Specify the user profile name.

Option (OPTION)

Specifies the option to be performed on the user certificate.
*CVTRMV
Convert the mapping information to use EIM and move the certificates themselves into LDAP. The existing certificates and mapping information will be removed from the user profile.

*RMV Remove the mapping information and certificates from the user profile.

*CVT Copy the mapping information to EIM and certificates into LDAP. The existing certificates and mapping information will remain in the user profile.

Examples
CVTUSRCERT USRPRF(JOHNSON) OPTION(*CVTRMV)
This command converts all digital certificates for user profile JOHNSON to use EIM for mapping information and LDAP for storing the certificates. The certificates and mapping information currently stored with the user profile will be removed.

Error messages
*ESCAPE Messages
CPF2204
User profile &1 not found.

CPF2213
Not able to allocate user profile &1.

CPF2225
Not able to allocate internal system object.

CPF222E
&1 special authority is required.

CPF4AB9
User certificate function not successful.
Data (DATA)

Where allowed to run:
  • Batch job (*BATCH)

Threadsafe: No

The Data (DATA) command must be used in an input stream to indicate the beginning of an inline data file. This input stream is read by a spooling reader. The Data (DATA) command also specifies what delimiter must be used to indicate the end of the data file. Inline data files exist only during this job, after the job is finished, they are destroyed. Unnamed inline files can be used only once in the job.

Restrictions
1. The DATA command cannot be run from a workstation.
2. The DATA command must have two slashes (//) in positions 1 and 2 of the data record.
3. Blanks can separate the slashes from the command name (//DATA).

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>FILE</td>
<td>Input file</td>
<td>Name, QINLINE</td>
<td>Optional, Positional 1</td>
</tr>
<tr>
<td>FILETYPE</td>
<td>File type</td>
<td>*DATA, *SRC</td>
<td>Optional, Positional 2</td>
</tr>
<tr>
<td>ENDCCHAR</td>
<td>Characters for end of data</td>
<td>Character value, //</td>
<td>Optional, Positional 3</td>
</tr>
<tr>
<td>IGCDTA</td>
<td>User specified DBCS data</td>
<td>*NO, *YES</td>
<td>Optional</td>
</tr>
</tbody>
</table>

Input file (FILE)

Specifies the name of the inline data file. This name is also specified in the program that processes the file.

QINLINE

The name of the inline data file is QINLINE. The file is processed as an unnamed inline file. An unnamed file can be processed if the program specifies QINLINE as the file name, or if the device file that specifies *YES on the Spool the data (SPOOL) parameter is opened for input. Unnamed inline files can be used only once by the job.

name

Specify the name of the inline data file used by one or more programs in the job. The file is connected to the program when the program opens the file by specifying its file name. Named inline data files can be accessed more than once by the job.
File type (FILETYPE)

Specifies whether the inline data following this command is put in the standard format for source files or in the data file format. The standard source file format is a sequence number (a 6-character source number) followed by the 6-character system date that goes before the data.

*DATA
   The inline data is not in the standard format for source files. The data file is passed to the program, which uses it in the data format.

*SRC
   The inline data is numbered in sequence; it is a source file that can be used to create another file or a program.

Characters for end of data (ENDCHAR)

Specifies a string of characters used to indicate the end of an inline data file. To be recognized, the character string must begin in position 1 of the record. If you specify a character string other than // (the default value) as the delimiter, all records up to the end-of-file record (the record containing the specified character string starting in column 1) are treated as data. This allows you to embed reader commands in the data stream. The end-of-file record for ENDCHAR values (which are not default values) is not put to the data file, and it is not checked to see if it is a valid reader command. It is used only to determine the end of the data stream and then it is discarded.

//
   The default value is two slashes. The command works the same way whether two slashes are coded into the parameter or the parameter itself is defaulted. Using the default, the slashes in positions 1 and 2 of a record (in either a data file or a source file) identify the first record beyond the file.

character-value
   A character string (up to 25 characters long and enclosed in apostrophes) can be entered to identify the last record in the file. The character string can contain both alphanumeric and special characters. If a character combination other than ‘//’ is specified on the Characters for end of data (ENDCHAR) parameter, reader commands can be safely embedded in the data. The reader ignores all other data while searching for the specified string, including reader commands.

User specified DBCS data (IGCDTA)

Specifies whether the inline data following this command may contain double-byte character set (DBCS) data.

*NO
   The inline file does not contain any DBCS data.

*YES
   The inline file may contain DBCS data.

Examples

Example 1: Inline Data File in Data File Format

//DATA   FILE(FILE1)

This command assigns the name FILE1 to the data that follows it, until an end of inline data condition is found (two slashes in positions 1 and 2).
Example 2: Specifying an End Character String

//DATA FILE(FILE2) ENDCCHAR('STOPIT')

This command assigns the name FILE2 to the data following it; the file continues until a record is found that contains the characters STOPIT in positions 1 through 6. This delimiter allows the //BCHJOB, //ENDBCHJOB, and //DATA commands and records with // in positions 1 and 2 to be embedded in an inline file.

Example 3: Specifying a File Containing DBCS Data

//DATA FILE(FILE3) IGCOTA(*YES)

This command assigns the name FILE3 to the data that follows it. This file can contain DBCS data.

Error messages

*ESCAPE Messages

CPF1753

Command cannot be run.
Declare CL Variable (DCL)

Where allowed to run:

- Batch program (*BPGM)
- Interactive program (*IPGM)

Threadsafe: Yes

The Declare CL Variable (DCL) command defines the Control Language (CL) program variables that are used in CL procedures. CL variables are used to store and update data and to receive parameters from another program on a call. CL variables are known by name only within the program that declares them. They cannot be used outside a CL procedure except when they are referred to by some commands (such as the DSPPGMVAR command) used for debugging programs. However, the value in the variable can be passed to another program as a parameter. If a variable is declared, but not referred to by another command in a CL procedure, the variable is not included in the program when it is compiled. Each DCL command defines the attributes of one CL variable and declares its name in the program in which it is used.

Each CL variable in a program must be identified by one of the two declare commands. The Declare File (DCLF) command declares CL variables for display device files and database files. The DCL command declares all other CL variables.

Restrictions: The DCL command is valid only within a CL procedure. All declare commands (DCL, COPYRIGHT, DCLF, and DCLPRCOPT) must follow the PGM (Program) command and must precede all other commands in the program. The four types of declare commands can be intermixed in any order.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>VAR</td>
<td>CL variable name</td>
<td>CL variable name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td>TYPE</td>
<td>Type</td>
<td>&quot;DEC, &quot;CHAR, &quot;LGL, &quot;INT, &quot;UINT, &quot;PTR</td>
<td>Required, Positional 2</td>
</tr>
<tr>
<td>STG</td>
<td>Storage</td>
<td>&quot;AUTO, &quot;BASED, &quot;DEFINED</td>
<td>Optional</td>
</tr>
<tr>
<td>LEN</td>
<td>Length of variable</td>
<td>Element list</td>
<td>Optional, Positional 3</td>
</tr>
<tr>
<td></td>
<td>Element 1: Length</td>
<td>Integer</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Decimal positions</td>
<td>Integer</td>
<td></td>
</tr>
<tr>
<td>VALUE</td>
<td>Initial value</td>
<td>Character value</td>
<td>Optional, Positional 4</td>
</tr>
<tr>
<td>BASPTR</td>
<td>Basing pointer variable</td>
<td>CL variable name</td>
<td>Optional</td>
</tr>
<tr>
<td>DEFVAR</td>
<td>Defined on variable</td>
<td>Element list</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Element 1: CL variable name</td>
<td>CL variable name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Position</td>
<td>1-32767, 1</td>
<td></td>
</tr>
<tr>
<td>ADDRESS</td>
<td>Address</td>
<td>Element list</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Element 1: CL variable name</td>
<td>CL variable name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Offset</td>
<td>0-32766, 0</td>
<td></td>
</tr>
</tbody>
</table>
**CL variable name (VAR)**

Specifies the CL variable to be declared in the CL procedure. The variable exists only within the program in which it is defined. It can be passed as a parameter on a call to another program, in which case it can be processed by the called program. The name must start with an ampersand (&).

This is a required parameter.

---

**Type (TYPE)**

Specifies the type of value contained in the CL variable to be declared. The value for this parameter cannot be specified by a CL variable.

This is a required parameter.

*DEC  A decimal variable that contains a packed decimal value.

*CHAR  A character variable that contains a character string value.

*LGL  A logical variable that contains a logical value of either ‘1’ or ’0’.

*INT  An integer variable that contains a signed binary value.

*UINT  An integer variable that contains an unsigned binary value.

*PTR  A pointer variable that contains an address.

---

**Storage (STG)**

Specifies the storage type of the variable. The value for this parameter cannot be specified by a CL variable.

**AUTO**  The storage for this variable is allocated in automatic storage.

**BASED**  The storage for this variable is based on the pointer variable specified on the **Basing pointer variable** (BASPTR) parameter. A based CL variable cannot be used unless the basing pointer variable has been set to a valid address.

**DEFINED**  The storage for this variable is provided by the CL variable specified on the **Defined on variable** (DEFVAR) parameter.

---

**Length of variable (LEN)**

Specifies the length of the CL variable to be declared. If the variable is a decimal value, the number of decimal digits to the right of the decimal point can be optionally specified. The value for this LEN parameter cannot be specified by a CL variable.
**Note:** If *PTR is specified for the TYPE parameter, you cannot specify a value for this parameter. Pointers have a fixed length of 16 bytes.

**Element 1: Length**

*length*  For character or integer CL variables, specify the number of bytes for the CL variable. For decimal CL variables, specify the maximum number of digits for the CL variable. The length cannot be greater than the maximum for this type of variable. The length must be either 2 or 4 for integer CL variables.

**Element 2: Decimal positions**

*decimal-positions*

This element is valid only for decimal variables. The length of the value in the variable includes the number of decimal positions in the value. The maximum length of the decimal value is 15 digits, including the digits to the right of the decimal point. Up to nine decimal positions can be specified. If nine decimal positions are specified, the value to the left of the decimal point can never be greater than 999,999 because only 6 of the 15 digits are left for the integer value.

If a length (in digits) is specified for a decimal variable and the number of decimal positions is not specified, 0 decimal positions is assumed.

The maximum lengths for each of the five types are:

- Decimal — 15 digits, 9 decimal positions
- Character — 32767 bytes
  
  **Note:** The initial value (specified for the VALUE parameter) of a CL variable can be no greater than 5000 characters.
- Logical — 1 byte
- Integer — 4 bytes
- Unsigned integer — 4 bytes

The default lengths for each of the five types are:

- Decimal — 15 digits, 5 decimal positions
- Character — 32 bytes
- Logical — 1 byte
- Integer — 4 bytes
- Unsigned integer — 4 bytes

**Note:** For decimal and character types, the default length is the same as the length of the initial value, if one is specified in the VALUE parameter.

---

**Initial value (VALUE)**

Specifies the initial value that is assigned to the CL variable when it is declared in the program. The value must be of the type specified by the TYPE parameter. If no value is specified, a character variable is set to blanks, a decimal, integer, or unsigned integer variable is set to a value of zero, and a logical variable is set to ‘0’. The value for the VALUE parameter cannot be specified by a CL variable.

The VALUE parameter may not be specified for *PTR CL variables, or CL variables declared with *DEFINED or *BASED specified for the Storage (STG) parameter.
If the name of the declared variable is specified for the PARM parameter of the PGM command in the same program in which the variable is declared, an initial value cannot be specified for the variable. In that case, the variable receives its value from the calling program.

---

**Basing pointer variable (BASPTR)**

Specifies the basing pointer for a CL variable declared with storage of *BASED.

*Note:* This parameter must be specified if *BASED is specified for the Storage (STG) parameter.

**CL-variable-name**

- Specify the name of a CL variable declared as TYPE(*PTR) which will serve as the basing pointer for the based CL variable being declared. This pointer must be initialized to a value before the based variable can be used.
- The name must start with an ampersand (&).

---

**Defined on (DEFVAR)**

Specifies the CL variable that the variable being declared is to be defined on.

*Note:* This parameter must be specified if *DEFINED is specified for the Storage (STG) parameter.

*Note:* A variable declared as STG(*DEFINED) cannot extend beyond the last byte of of the CL variable that it is defined on.

**Element 1: CL variable name**

**CL-variable-name**

Specify the name of the CL variable that the variable being declared is defined on.

**Element 2: Starting position**

1-32767

Specify the starting position of the variable being declared from the beginning of the defined-on variable.

---

**Address (ADDRESS)**

Specifies the initial address for a CL variable declared with *PTR as the TYPE value.

*Note:* A value cannot be specified for this parameter unless the variable being declared is a pointer variable and *AUTO is specified for the STG parameter.

**Element 1: CL variable name**

**CL-variable-name**

- Specifies the name of a CL variable which is to be the initial address for the pointer variable.
- The name must start with an ampersand (&).
Element 2: Offset

0  The pointer variable is set to the first byte of the CL variable being addressed.
0-32766  Specify the number of bytes from the beginning of the variable being addressed that the pointer is to be set.

Examples

Example 1: Specifying the CL Variable Length

DCL &ABLE *DEC LEN(5 2)

This command declares a CL variable named &ABLE that contains a decimal value. The value can never be greater than 999.99 because LEN specifies up to 5 digits, of which two are to the right of the decimal point. Because the VALUE parameter was not specified, and it is a numeric value, &ABLE is set to a value of zero (000.00).

Example 2: Specifying a Logical Value

DCL &SWITCH *LGL

This command declares a CL variable named &SWITCH to contain a logical value. Because the type parameter specifies logical, the variable is one character long and it is set to '0'.

Example 3: Specifying Initial Value of CL Variable

DCL &FILNAM *CHAR VALUE(FILEA)

This command declares a CL variable named &FILNAM whose value is FILEA. Because the value contains 5 characters and the LEN parameter was not specified, the length of the variable is also 5 characters.

Example 4: Specifying Defined CL Variables

DCL &QUALOBJ *CHAR LEN(20)
DCL &OBJ *CHAR LEN(10) STG(*DEFINED) DEFPAR(&QUALOBJ 1)
DCL &LIB *CHAR LEN(10) STG(*DEFINED) DEFPAR(&QUALOBJ 11)

The first DCL command declares a 20-character variable in the program’s automatic storage. The second DCL command declares a variable named &OBJ which refers to the first 10 characters of the &QUALOBJ variable. The last DCL command declares a variable named &LIB which can be used to reference the last 10 characters of the &QUALOBJ variable.

Example 5: Specifying Pointer CL Variables

DCL &CHAR *CHAR LEN(10)
DCL &PTR *PTR ADDRESS(&CHAR)

The second DCL command declares a pointer variable which is initialized to point to the &CHAR variable in the program’s automatic storage.

Example 6: Specifying Based CL Variables

DCL &PTR *PTR
DCL &CHAR *CHAR LEN(10) STG(*BASED) BASPTR(&PTR)
The second DCL command declares a character variable which is found at the location addressed by the &PTR variable. Before the &CHAR variable can be used, the &PTR variable must be initialized to a valid address by using the %ADDRESS built-in function.

Example 7: Specifying Defined Pointer CL Variables

DCL &CHAR *CHAR LEN(48)
DCL &PTR *PTR STG(*DEFINED) DEFVAR(&CHAR 17)

The second DCL command declares a pointer variable in bytes 17 through 32 of the variable &CHAR.

Error messages

None
Declare File (DCLF)

Where allowed to run:
- Batch program (*BPGM)
- Interactive program (*IPGM)

Threadsafe: Yes

The Declare File (DCLF) command declares one file (by name) to a Control Language (CL) program. Up to five DCLF commands are allowed in a CL procedure. Each DCLF command specifies the name of a display file or database file, the file record formats to be used in the program, and an optional open file identifier which is used to uniquely identify the declared instance of the file within the CL procedure. Multiple DCLF commands can reference the same file, so long as the value specified for the Open file identifier (OPNID) parameter is unique. Following the DCLF command for a file, the CL procedure can contain data manipulation commands. For display files, the following commands can be used to send data to a workstation and receive data from a workstation: Send File (SNDF), Receive File (RCVF), Send/Receive File (SNDRCVF), End Receive (ENDRCV), and Wait (WAIT). For database files, the RCVF command can be used to read records from the file.

When the CL procedure is compiled, a CL variable is automatically declared for each field in each record format used in the program. If the file is a record-level database file, the record format contains one field with the name of that record format. If the value specified for the OPNID parameter is *NONE, the variable name is the field name prefixed with an ampersand (&). If the OPNID parameter value is not *NONE, the variable name is the field name prefixed with an ampersand (&), the value specified for the OPNID parameter, and an underscore.

For example, if a declared file has a record format with field CUSTNAME and the open file identifier specified on the DCLF command was FILE1, the declared variable would be:

&FILE1_CUSTNAME

The attributes of each declared field are the same as the attributes of the field in the file record format. Fields defined in the record format as numeric are defined as decimal variables. Indicators defined in the referenced file record format are declared as logical variables with a variable name in the form INNn, where ‘nn’ is the indicator number.

Variables automatically declared by the DCLF command can be used in the program the same as the variables declared by a DCL command. For example, indicators can be used in expressions and IF statements because they are declared as logical variables.

The content of the variables, not the variable names, are seen by the user; the display shows one, some, or all of the fields in the record format that can be filled in by the user. DDS determines the display format.

Restrictions:
- This command is valid only within CL procedures. All declare commands (DCL, COPYRIGHT, DCL, and DCLPRCOPT) must follow the PGM (Program) command and must precede all other commands in the program. The four types of declare commands can be intermixed in any order.
- The file must either be a database file with only one record format or be a display file.
- The file cannot be a mixed file, even if only display devices are defined for that mixed file.
- The database file can be either physical or logical, and can be either field-level or nonfield level.
- The referenced file must exist before the program is created.
Because CL variables are automatically declared for each field in a referenced file’s record formats, the following restrictions apply:

- If the file is changed (and the file description specifies that level checking is to be performed), the CL procedure must be recompiled to match the new file description. More information on level checking is in the Database information in the iSeries Information Center at http://www.ibm.com/eserver/iseries/infocenter and the Application Display Programming book, SC41-5715.
- If any field name is defined in more than one record format of the display file, the attributes in each record format for the commonly named field must match.
- Any CL variable declared in the program by a DCL command with the same name as an automatically declared CL variable (for a referenced field) must also have the same attributes specified in DDS for the referenced field.
- The variables used in the file must have data types supported for CL variables. Fields defined as packed decimal format or zoned decimal format are declared as decimal variables. Fields defined as binary format are declared as decimal variables by default, but will be declared as integer variables only if all of the following conditions are true:
  - *INT is specified for the Declare binary fields (DCLBINFLD) parameter.
  - The field size is less than 10 digits.
  - The field precision is 0.
- The variables used in the file must have attributes that are valid for CL variables. For decimal variables, the limits are 15 digits and 9 decimal positions. For variables with more than 15 digits, CL will declare a character variable that is the number of digits, divided by 2, plus 1 \((n / 2 + 1)\) in length. A CPI0306 message will be issued stating that the conversion has taken place, which can be seen in the program listing. For character variables, the limit is 32767 bytes.

Additional Considerations:

File processing is handled differently in CL procedures, depending on whether the file specified in the DCLF command is a display file or a database file.

The following statements apply if the file is a display device file at compile time:

- The file must be a display device file at run time.
- The program may contain any or all of the following commands: SNDF, RCVF, SNDRCVF, ENDRVC, and WAIT.
- The file is opened for input and output.

The following statements apply if the file is a database file at compile time:

- The file must be a database file with a single record format at run time.
- The program may contain only RCVF commands; SNDF, SNDRCVF, ENDRVC, and WAIT commands are not allowed.
- The file is opened for input only.
- The file is implicitly opened when the RCVF command is run, not by using the Open Database File (OPNDBF) command.
- The file is implicitly closed when the CL procedure ends, not by using the Close File (CLOF) command.
### File (FILE)

Specifies the file to be used by the CL procedure.

This is a required parameter.

**Qualifier 1: File**

name Specify the name of the file.

**Qualifier 2: Library**

*LIBL All libraries in the library list for the current thread are searched until the first match is found.

*CURLIB The current library for the job is used to locate the file. If no current library entry exists in the library list, QGPL is used.

name Specify the library where the file is located.

### Record format (RCDFMT)

Specifies the names of one or more record formats contained in the file. These record formats are used by the Send File (SNDF), Receive File (RCVF), and Send/Receive File (SNDRCVF) commands in the CL procedure. Database files can be processed only by RCVF. CL variable names cannot be specified in RCDFMT; only names of record formats can be used. For every field and indicator in each record format specified in RCDFMT, one CL variable is automatically declared in the program.

**Note:** A physical file can contain only one record format. A logical file which has multiple record formats defined in DDS may be used if it is defined over only one physical file member. If the physical file contains more than one record format, an error message is sent and the compile procedure fails.

**Single values**

*ALL Every record format in the file, up to a maximum of 99, is to have its fields declared in the CL program as variables. If there are more than 99 record formats in the file, only the first 99 are used.
Other values (up to 50 repetitions)

`name` Specify the name of the file record format whose fields are to be declared as variables in the CL procedure. CL variables cannot be used to specify the names.

Open file identifier (OPNID)

Specifies the open file identifier to be associated with the file specified for the `File (FILE)` parameter. This identifier must be unique for all files declared in the CL program.

**NONE**

The file does not have an open file identifier. Only one file can be declared in a CL procedure with *NONE as the open file identifier.

`simple-name`

Specify the name to be used as the open file identifier for the file.

Allow variable length fields (ALWVARLEN)

Specifies whether variable length fields are allowed in record formats.

**NO** Variable length fields are not allowed in record formats.

**YES** Variable length fields are allowed in record formats. CL variables declared for variable-length fields are handled as type *CHAR with length equal to 2 bytes plus the maximum field length. Following a RCVF on a variable-length field, the first 2 bytes in the CL variable contain the length of the data. The data received from the field is padded on the right with blanks to the maximum length allowed (32765 bytes).

Allow field value of null (ALWNULL)

Specifies whether a field value of null is allowed.

**NO** Values of null are not allowed. For each field containing a null value at RCVF time, a diagnostic message is sent with a single escape message for the entire record. Default values are placed in the CL variables.

**YES** Values of null are allowed.

Allow graphic fields (ALWGRAPHIC)

Specifies whether graphic data fields are allowed in record formats.

**NO** Record formats cannot contain graphic data fields. A diagnostic message is sent at compile time if graphic data fields are supported in the file.

**YES** Record formats can contain graphic data fields. CL variables declared for graphic data fields are handled as type *CHAR with length equal (in bytes) to the graphic data field length.
Declare binary fields (DCLBINFLD)

Specifies whether variables declared for binary fields in the record format should be packed decimal or integer.

*D DEC CL variables declared for binary fields in the record format will use TYPE(*DEC).

*INT CL variables declared for binary fields, with precision of zero and a length of 9 or less in the record format, will use TYPE(*INT).

Examples

Example 1: Declaring Fields of All Record Formats as Variables

```
DCLF FILE(ABLE) RCDFMT(*ALL)
```

This command specifies that the file named ABLE is used by the CL program to pass data between the user and the program. Because no library was specified, the library list is used to locate the file. All the fields and indicators in all the record formats are automatically declared as variables, and data from any field in any record format (up through the first 99) in the file can be passed between the program and the user.

Example 2: Using Multiple Record Formats

```
DCLF FILE(BAKER) RCDFMT(REC2 REC6)
```

Display file BAKER is used by the CL procedure to pass data between the user and the program. Assuming the library qualifier for FILE defaults to *LIBL, the library list is used to locate the file. Both the REC2 and REC6 record formats are used.

Example 3: Using an Open File Identifier

```
DCLF FILE(MYLIB/CHARLES) OPNID(CTLFILE1)
```

File CHARLES in library MYLIB is used by the the CL procedure to read records from the database file. If the record format contains a field named CUSTNUMBER, the following variable will be declared:

```
&CTLFILE1_CUSTNUMBER
```

Error messages

None
Declare Processing Options (DCLPRCOPT)

Where allowed to run:
- Batch program (*BPGM)
- Interactive program (*IPGM)

Threadsafe: Yes

The Declare Processing Options (DCLPRCOPT) command lets you define compiler processing options.

Restrictions:
- This command is valid only within CL procedures. All declare commands (DCL, COPYRIGHT, DCLF, and DCLPRCOPT) must follow the PGM (Program) command and must precede all other commands in the program. The four types of declare commands can be intermixed in any order.
- Only one DCLPRCOPT command is allowed by the CL compiler; if more than one are specified, message CPD0323 is sent and the compile fails.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>SUBRSTACK</td>
<td>Subroutine stack depth</td>
<td>20-9999, 99</td>
<td>Optional, Positional 1</td>
</tr>
</tbody>
</table>

Subroutine stack depth (SUBRSTACK)

Specifies how many entries you want to allow on the subroutine stack. Each time a CALLSUBR (Call Subroutine) command is run, an entry is added on the subroutine stack. The entry is removed when a RTNSUBR (Return from Subroutine) or ENDSUBR (End Subroutine) command is run. The subroutine stack can have multiple entries when CALLSUBR commands are run from within a subroutine; a subroutine can invoke another subroutine or recursively invoke itself.

- **99** The maximum number of subroutine stack entries allowed when this CL program is run is 99.
- **20-9999** Specify the maximum number of subroutine stack entries allowed when this CL program is run.

Examples

DCLPRCOPT SUBRSTACK(50)

This command sets the maximum number of subroutine stack entries to 50. When the CL program is run, if the subroutine stack depth exceeds 50, escape message CPF0822 will be sent.
Error messages

None
Decompress Object (DCPOBJ)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Decompress Object (DCPOBJ) command allows you to permanently decompress programs, panel groups, menus, display files, printer files, modules, and service programs.

- Compressed objects use less storage space than decompressed objects. When a compressed object is used or a compressed program is called, a decompressed version of the object automatically becomes available to the user.
- Decompressed objects use the system storage space allocated to them and are in a final, ready-to-use state.
- Temporarily Decompressed Objects are temporarily decompressed copies of compressed objects. The system allocates storage space for the temporary copies until the system or the user determines that the temporary storage space needs to be reclaimed. Temporary storage is automatically reclaimed when:
  - The RCLTMPSTG command is run
  - The next initial program load (IPL) is run
  - The object is used often enough to cause the system to permanently decompress it

When an object is permanently decompressed, the compressed version of the object is destroyed as well as any temporary forms of the object; however, compressed versions remain intact as long as the objects are temporarily decompressed.

Restriction: The user must have *USE authority to the objects specified on the command and execute authority to the library containing the objects.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>OBJ</td>
<td>Object</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Object</td>
<td>Generic name, name, *ALL</td>
<td></td>
</tr>
<tr>
<td>PGMOPT</td>
<td>Program option</td>
<td>*ALL, *INS</td>
<td>Optional</td>
</tr>
</tbody>
</table>

Object (OBJ)

Specifies the name and library of the object to be decompressed.

This is a required parameter.

The possible values are:
*ALL All objects in the specified library of the object type specified on the Object type prompt (OBJTYPE parameter) are decompressed.

generic*-object-name
Specify the generic name of the object to be decompressed. A generic name is a character string that contains one or more characters followed by an asterisk (*).

object-name
Specify the name of the object to be decompressed.

The possible library values are:

*LIBL All libraries in the library list for the current thread are searched until the first match is found.

*USRLIBL
If a current library entry exists in the library list for the current thread, the current library and the libraries in the user portion of the library list are searched. If there is no current library entry, only the libraries in the user portion of the library list are searched.

*CURLIB
Only the current library is searched. If no current library entry exists in the library list, QGPL is used.

*ALL All libraries in the system, including QSYS, are searched.

*ALLUSR
All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

#CGULIB #DSULIB #SEULIB
#COBLIB #RPGLIB
#DFULIB #SDALIB

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered user libraries and are also searched:

QDSNX QRCllx xxx QUSRJJS QUSRVxRxMx
QGPL QSRVAGT QUSRINFSKR
QGPL3B QSYS2 QUSRNOTES
QMGT C QSYS2xxxxx QUSROND
QMGT C2 QS36 F QUSRPOS G
QM PDATA QUSER3B QUSRPOSSA
QM PDATA QUSRADSM QUSRPOYMSVR
QMMPROC QUSRBBM QUSRPOQG
QPFRDATA QUSRDIRCL QUSR SYS
QRC L QUSRDIRDB QUSRVI

1. ‘xxxxx’ is the number of a primary auxiliary storage pool (ASP).

2. A different library name, in the format QUSRVxRxMx, can be created by the user for each previous release supported by IBM to contain any user commands to be compiled in a CL program for the previous release. For the QUSRVxRxMx user library, VxRxMx is the version, release, and modification level of a previous release that IBM continues to support.

library-name
Specify the name of the library to be searched.
Object type (OBJTYPE)

Specifies the type of object to be decompressed. You can specify *ALL, or you can specify one or more of the other possible values.

This is a required parameter.

The possible values are:

*ALL  All compressed menus, panel groups, display and printer device files, programs, modules, and service programs with the name and library specified on the Object prompt (OBJ parameter) are decompressed.

*FILE  Display and printer device files with the name and library specified on the Object prompt (OBJ parameter) are decompressed.

*MENU  Menus with the name and library specified on the Object prompt (OBJ parameter) are decompressed.

*MODULE  Modules with the name and library specified on the Object prompt (OBJ parameter) are decompressed.

*PGM  Programs with the name and library specified on the Object prompt (OBJ parameter) are decompressed.

*PNLGRP  Panel groups with the name and library specified on the Object prompt (OBJ parameter) are decompressed.

*SRVPGM  Service programs with the name and library specified on the Object prompt (OBJ parameter) are decompressed.

Program option (PGMOPT)

Specifies whether the entire program or service program or only the instruction stream is decompressed. This parameter is valid only when *PGM, *SRVPGM, or *ALL is specified on the Object type prompt (OBJTYPE parameter).

The possible values are:

*ALL  The entire program or service program (instruction stream and observability tables) is decompressed.

*INS  Only the instruction stream of the program or service program is decompressed.

Examples

DCPOBJ OBJ(QGPL/*ALL) OBJTYPE(*FILE)

This command decompresses all compressed display and printer files in library QGPL.
Error messages

*ESCAPE Messages

CPF2110
   Library &1 not found.

CPF2113
   Cannot allocate library &1.

CPF2176
   Library &1 damaged.

CPF3B01
   Cannot compress or decompress object &1 in &2.

CPF3B02
   Cannot compress or decompress file &1 in &2.

CPF3B05
   No objects decompressed.

CPF3B06
   &1 objects decompressed; &3 not decompressed; &8 not included.

CPF3B08
   Cannot allocate object &1 in &2.

CPF8108
   Device file or save file &4 in &9 damaged.

CPF812E
   Module &4 in &9 damaged.

CPF8129
   Program &4 in &9 damaged.

CPF813D
   Service program &4 in &9 damaged.

CPF8150
   Panel group &4 in &9 damaged.

CPF8151
   Menu &4 in &9 damaged.

CPF9570
   Error occurred creating or accessing debug data.

CPF9802
   Not authorized to object &2 in &3.

CPF9803
   Cannot allocate object &2 in library &3.

CPF9804
   Object &2 in library &3 damaged.

CPF9806
   Cannot perform function for object &2 in library &3.

CPF9807
   One or more libraries in library list deleted.

CPF9808
   Cannot allocate one or more libraries on library list.
CPF9811
  Program &1 in library &2 not found.

CPF9812
  File &1 in library &2 not found.

CPF9821
  Not authorized to program &1 in library &2.

CPF9822
  Not authorized to file &1 in library &2.

CPF9838
  User profile storage limit exceeded.
Remove Link (DEL)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Remove link (DEL) command removes the link to the specified object. If this is the only hard link to the object, the object is removed when no longer in use. The object can be removed even if a symbolic link to it exists. The symbolic link remains until it is removed.

This command is an alias for the Remove link (RMVRLNK) command and can also be issued using the following alternative command names:
• ERASE
• RMVRLNK

For more information about integrated file system commands, see the Integrated file system information in the iSeries Information Center at http://www.ibm.com/eserver/iseries/infocenter.

Restrictions:
1. In the "root" (/), QOpenSys, and user-defined file systems, the user must have write, execute (*WX) authority to the directory containing the object. If a hard link is to be unlinked, the user must also have object existence (*OBJEXIST) authority to the object.
2. In the QDLS file system, the user must have all (*ALL) authority to the object and execute (*X) authority to the parent directory.
3. The user must have *X authority to each directory in the path.
4. See the iSeries Security Reference, SC41-5302 book for the authority requirements for other file systems.
5. A user cannot unlink an object within a "root" (/), QOpenSys, or user-defined file system directory that has the "restricted rename and unlink" attribute set on (this attribute is equivalent to the S_ISVTX mode bit) unless one or more of the following are true:
   a. The user is the owner of the object.
   b. The user is the owner of the directory.
   c. The user has all object (*ALLOBJ) special authority.
6. A directory cannot be unlinked.
7. The link to a file cannot be removed if the file is a DataLink column in an SQL table and where a row in that SQL table references this file.
8. The restrictions listed above are for the i5/OS objects of the types *DDIR, *DSTMF, *SOCKET, *STME, and *SYMLNK.

QSYS.LIB and independent ASP QSYS.LIB File System Differences
1. If this command is to be used to remove links for an object that is in these file systems, additional restrictions may apply. To identify these restrictions, see the delete command for the object to be removed. In general, the name of this command is formed using the i5/OS object type value, from the character * is removed, and add the verb DLT to the beginning. For example, to delete an alert table, which has the object type value of *ALRTBL, see the Delete Alert Table (DLTALRTBL) command for any additional restrictions.
   However, there are exceptions to this rule. For example, to delete a compiler unit, which has the object type value of *MODULE, see the Delete Module (DLTMOD) command for any additional restrictions.
For a description of the object types, see the CL concepts and reference topic in the iSeries Information Center at http://www.ibm.com/eserver/iseries/infocenter.

2. In these file systems, libraries and database files cannot be deleted using the Remove Link (RMVLNK or alias DEL or ERASE) command. However, these objects can be deleted using the Remove Directory (RMVDIR or alias RMDIR or RD) command.


QDLS File System Differences
1. If this command is to be used to remove links for an object that is in this file system, additional restrictions may apply. To identify these restrictions, see the description of the Delete Document Library Object (DLTDLO) command.

### Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>OBJLNK</td>
<td>Object link</td>
<td>Path name</td>
<td>Required, Positional 1</td>
</tr>
</tbody>
</table>

### Object link (OBJLNK)

Specifies the path name of the object to unlink. Multiple links can be removed with a name pattern.

The object path name can be either a simple name or a name that is qualified with the name of the directory in which the object is located. A pattern can be specified in the last part of the path name. An asterisk (*) matches any number of characters and a question mark (?) matches a single character. If the path name is qualified or contains a pattern, it must be enclosed in apostrophes.

For more information on specifying path names, refer to "Object naming rules" in the CL concepts and reference topic in the iSeries Information Center at http://www.ibm.com/eserver/iseries/infocenter.

### Examples

The alternative command name for DEL is RMVLNK. The following examples use the alternative command name, but DEL can be replaced directly for RMVLNK in all of them.

**Example 1: Removing an Object Link**

```bash
RMVLNK OBJLNK('PAY')
```

This command removes a link named PAY.

### Error messages

*ESCAPE Messages*
CPFA085
   Home directory not found for user &1.

CPFA093
   Name matching pattern not found.

CPFA09C
   Not authorized to object. Object is &1.

CPFA0A1
   An input or output error occurred.

CPFA0A7
   Path name too long.

CPFA0A9
   Object not found. Object is &1.

CPFA0AB
   Operation failed for object. Object is &1.

CPFA0B1

CPFA0B2
   No objects satisfy request.

CPFA0BD
   &1 links removed. &2 links failed.
IBM Systems - iSeries: i5/OS Commands Starting with CRTMSGF (Create Message File)
Dependent Definition (DEP)

The Dependent Definition (DEP) command definition statement defines a required relationship between parameters and parameter values that must be checked. This relationship can refer to either the specific value of a parameter or parameters, or to the required presence of parameters.

DEP statements provide a second level of parameter syntax checking for a CL command. The first level is provided by the PARM, QUAL, and ELEM statements that define the type of value (like *NAME or *DATE), as well as parameter value restrictions such as a the range of valid values or a list of valid values. DEP statements allow you to verify that combinations of parameter values are syntactically correct within the command string.

DEP statements can only check the first value of a specified parameter. If you want to do syntax checking for a parameter that accepts a list of values or a parameter with multiple elements or qualifiers, a validity checking exit program can be written to do a third level of syntax check for the command string. Doing syntax checking in the PARM, QUAL, ELEM or DEP statements, or in a validity checking program, can remove or greatly simplify parameter syntax checking code in your command processing program.

If a parameter has a default value and the parameter is not specified, the checking differs depending on whether the DEP statement is performing a specification check or a relational check. If a specification check is made on an unspecified parameter (checking for the presence of a value for that parameter), the system assumes that no value was specified, and the default value is not used. If a relational check is made on an unspecified parameter, the default value is used as the parameter value in the relational check.

<table>
<thead>
<tr>
<th>Parameters</th>
</tr>
</thead>
</table>

### Parameters Table

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CTL</strong></td>
<td>Controlling conditions</td>
<td>Single values: *ALWAYS Other values: Element list</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Element 1: Keyword or keyword reference</td>
<td>Character value</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 3: Value or keyword reference</td>
<td>Character value</td>
<td></td>
</tr>
<tr>
<td><strong>PARM</strong></td>
<td>Dependent parameter</td>
<td>Values (up to 25 repetitions): Element list</td>
<td>Required, Positional 2</td>
</tr>
<tr>
<td></td>
<td>Element 1: Keyword or keyword reference</td>
<td>Character value</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 3: Value or keyword reference</td>
<td>Character value</td>
<td></td>
</tr>
</tbody>
</table>

© Copyright IBM Corp. 1998, 2006
<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>NBRTRUE</td>
<td>Number of true dependencies</td>
<td>Single values: *ALL</td>
<td>Optional, Positional 3</td>
</tr>
<tr>
<td></td>
<td></td>
<td>Other values: *Element list</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Number to be true</td>
<td>0-25</td>
<td></td>
</tr>
<tr>
<td>MSGID</td>
<td>Message identifier</td>
<td>Name, *NONE</td>
<td>Optional, Positional 4</td>
</tr>
</tbody>
</table>

### Controlling conditions (CTL)

Specifies the controlling conditions that must be true before the parameter dependencies defined in the PARM statement must be true. The first keyword specified identifies the controlling parameter. The controlling condition can be specified by a keyword name only, or by a keyword name and a test relationship that determines whether the controlling condition requires the presence of the parameters it depends on. The relationship between the controlling parameter and a specified value can be tested to determine if the condition specified is met. If it is, the parameters that the controlling parameter depends on must meet the requirements specified in the PARM and NBRTRUE keywords.

#### Single values

*ALWAYS

The parameter dependency is always checked, regardless of the form of the command.

#### Other values

**keyword-name**

Specify the keyword name of the parameter for which a value must be specified to control dependency. The keyword name is the name of the parameter that was specified by the **Keyword (KWD)** parameter on the PARM statement defining it. If the keyword was specified, the parameter dependency is checked. The keyword name cannot refer to a command parameter defined with TYPE(*NULL).

**&keyword-name relational-operator value**

Specify the keyword name of the controlling parameter followed by a relational operator (such as *LE or *EQ) and a value to be tested. If the tested condition is met, the parameters that the controlling parameter depends on must meet the requirements specified for the PARM keyword. The value must be no longer than 32 characters. The keyword name cannot refer to a command parameter defined with TYPE(*NULL).

If the value being tested against has been specified as a special value or single value, using the SPCVAL parameter or the SNGVAL parameter of the PARM statement, the to-value must be used rather than the from-value.

The keyword name must be preceded by an ampersand (&) to indicate that the value of the keyword is tested if the relational operator and value are specified; the ampersand must not be used if the relational operator and value are not specified.

**(&keyword-name relational-operator &keyword-name)**

Specify the keyword name of the controlling parameter followed by a relational operator (such as *EQ) and the keyword name of another parameter whose value is compared with the value of the controlling parameter. The keyword names cannot refer to command parameters defined with TYPE(*NULL) or PASSVAL(*NULL).
Dependent parameter (PARAM)

Specifies the parameter dependencies that must be tested if the controlling conditions defined by the CTL parameter are true. The dependencies can be the names of one or more parameters that are tested for their presence, or one or more test relationships of keyword values to other keyword values or constant values. A maximum of 25 parameter dependencies can be specified for this parameter. Keyword names cannot refer to command parameters defined with TYPE(*NULL).

**keyword-name**

Specify the keyword name of each parameter that must have a value specified for it.

**&keyword-name relational-operator value**

Specify the keyword name of each parameter followed by a relational operator and a value to be tested. An ampersand must precede the keyword name to indicate that the value of the keyword is tested. The value must be no longer than 32 characters.

If the value being tested against has been specified as a special value or single value, using the Special values (SPCVAL) parameter or the Single values (SNGVAL) parameter of the PARM statement, the to-value must be used rather than the from-value.

**&keyword-name relational-operator &keyword-name**

Specify the keyword name of one parameter followed by a relational operator and the keyword name of another parameter whose value is compared with the value of the first parameter. The keyword names cannot refer to command parameters defined with PASSVAL(*NULL).

Number of true dependencies (NBRTRUE)

Specifies the number of parameter dependencies (defined in the PARM parameter on this DEP statement) that must be true. Otherwise, a diagnostic message (defined in the MSGID parameter on this DEP statement) is sent and the command is not run.

CL variables cannot be coded for either element of this parameter.

**Single values**

**ALL** All the parameter dependencies must be true. This is the same as specifying NBRTRUE(*EQ n), where n is the number of parameter dependencies defined in the PARM parameter.

**Element 1: Relational operator**

**relational-operator**


**Element 2: Number to be true**

**0-25** Specify the number of parameter dependencies that must be true to satisfy the specified relationship.
Message identifier (MSGID)

Specifies the diagnostic message that is to be sent to the user if the logical expression specified by the NBRTRUE parameter evaluates as false.

*NONE

No specific diagnostic message is sent. Instead, generic message CPD0150 is sent. Depending on the number of parameters on the command, it can be very difficult to determine the cause of the interparameter syntax error from the generic message text.

message-identifier

Specify the message identifier of the diagnostic message sent to the user.

Messages whose identifiers begin with the 3-character prefixes CPF or CPD are retrieved from the IBM-supplied message file QCPFMSG. All other messages specified here are retrieved from the message file identified by the MSGF parameter on the CRTCMD command which is used to create the command being defined with these dependencies. Variables cannot be coded for this parameter.

Examples

Example 1: Checking the Presence of a Parameter

```
DEP  CTL(&TYPE +EQ LIST)  PARM(ELEMLIST)
```

If TYPE(LIST) is specified, the ELEMLIST parameter must be specified. If TYPE(LIST) and no value is specified for the ELEMLIST parameter, generic diagnostic message CPD0150 is sent and the command is not run.

Example 2: Checking the Presence of Multiple Parameters

```
DEP  CTL(FILE)  PARM(VOL LABEL) +
     NBRTRUE(*EQ 2)  MSGID(USR1234)
```

If the FILE parameter is specified, both the VOL and LABEL parameters must be specified. If only one of the VOL and LABEL parameters have a value specified, or if neither parameter is specified, diagnostic message USR1234 is sent and the command is not run. Command analyzer will look for message USR1234 in the message file specified for the MSGF parameter on the CRTCMD command.

Example 3: Checking for Mutually Exclusive Parameters

```
DEP  CTL(*ALWAYS)  PARM(J1 J2)  NBRTRUE(*EQ 1)
```

A value must be specified for one (and only one) of the J1, D, and J2 parameters. If zero or two or three of these parameters are specified, generic diagnostic message CPD0150 is sent and the command is not run.

Example 4: Checking One or More Conditions are True

```
DEP  CTL(&LIB *EQ MYLIB) +
     PARM((&PASSWORD *EQ XYZ5) (&USRPRF *EQ BOBJ)) +
     NBRTRUE(*GE 1)  MSGID(MSG1001)
```

If the LIB parameter value is MYLIB, the PASSWORD parameter value must be XYZ5, or the USRPRF parameter value must be BOBJ, or both PASSWORD(XYZ5) and USRPRF(BOBJ) must be specified. If LIB(MYLIB) and neither of the dependency conditions specified are true, diagnostic message MSG1001 is sent and the command is not run.

Example 5: Checking for a Conditionally Required Parameter
Three related interparameter checks will be made:

1. If the OUTPUT parameter has a value of *OUTFILE, the OUTFILE parameter must have a non-blank value specified. Otherwise, message CPD9861 is sent. Since the message identifier starts with 'CPD', the operating system will look for the message in message file QCPFMSG. Assuming the OUTFILE parameter is a qualified object name, only the value of the first QUAL will be checked.

2. If the OUTPUT parameter has a value other than *OUTFILE, the OUTFILE parameter must be blank, assuming that the OUTFILE parameter was coded as MIN(0) and with no default (DFT) value. Otherwise, message CPD9862 is sent.

3. If the OUTMBR parameter has any value other than *FIRST, the OUTFILE parameter must have a non-blank value specified. Otherwise, message CPD9867 is sent. Assuming the OUTMBR parameter is a list of two elements, only the value of the first ELEM will be checked.

Error messages

None
Deallocation Object (DLCOBJ)

Where allowed to run: All environments (*ALL)

Threadsafe: Conditional

The Deallocation Object (DLCOBJ) command releases the allocations of the specified objects. The objects, allocated earlier by one or more Allocate Object (ALCOBJ) commands, are freed for use by other jobs, or threads. If the DLCOBJ command is used when a lock does not exist, no error occurs.

If the DLCOBJ command is not used, the objects may be automatically deallocated. Allocated job-scoped locks are automatically released when the job ends. Allocated thread-scoped locks are automatically released when the thread ends. If a thread received a job-scoped lock, the job will continue to hold that lock after the requesting thread ends. Lock-space-scoped locks are not automatically released.

The DLCOBJ command should not be issued for an object that was not explicitly allocated by the ALCOBJ command. If the DLCOBJ command is used this way, internal locks on the object are released, making the object capable of being deleted.

To release more than one lock for an object with a single DLCOBJ command, the object name, type, and lock state must be repeated in the list for each lock you want to release.

NOTES:
1. When deallocating distributed data management (DDM) files and distributed files, additional time is required for the command to complete because of the time required for establishing communication and for deallocating files on remote systems.
2. Allocating an object by specifying *LIBL for the object’s library, changing the thread’s library list, and then attempting to deallocate the object by specifying *LIBL for the object’s library can result in issuing the deallocate against the wrong object. This could release internal locks.

Restrictions:
1. This command cannot be used to deallocate a device description, *DEVD, for an advanced program-to-program communications (APPC) device or for an intrasystem (INTRA) device.
2. This command can be used to deallocate only the following database *FILE types:
   • Physical files
   • Logical files
   • Distributed files
     This deallocates the piece of the file on each node in the node group.
   • DDM files
     This deallocates both the DDM file on the local system and the file on the remote system that is identified in the DDM file.
3. In multithreaded jobs, this command is not threadsafe for distributed files. This command is also not threadsafe for distributed data management (DDM) files of type *SNA.
## Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>OBJ</td>
<td>Object specifications</td>
<td>Values (up to 50 repetitions): <em>Element list</em></td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td>Element 1: Object</td>
<td>Qualified object name</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qualifier 1: Object</td>
<td>Name</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Element 4: Member, if data base file</td>
<td>Name, *FIRST</td>
<td></td>
<td></td>
</tr>
<tr>
<td>SCOPE</td>
<td>Lock scope</td>
<td>*JOB, *THREAD, *LCKSPC</td>
<td>Optional</td>
</tr>
</tbody>
</table>

## Object specifications (OBJ)

Specifies the qualified name of one or more objects that are deallocated from the job, thread, or lock space, the type of each object specified, the lock state of each object, and the member name (if the object is a database file or DDM file).

Only some object types can be specified on the Deallocate Object (DLCOBJ) command. Of these, some cannot use all of the lock states.

This is a required parameter.

You can specify 50 values for this parameter.

**Element 1: Object**

Qualifier 1: Object

.name Specify the name of the object.

Qualifier 2: Library

*LIBL All libraries in the thread’s library list are searched until a match is found.

*CURLIB The current library for the thread is used to locate the object. If no library is specified as the current library for the thread, the QGPL library is used.

.name Specify the name of the library where the object is located.
Element 2: Object type

*object-type*

Specify the type of object to be deallocated. Refer to the figure Valid Lock States by Object Type for more information.

Element 3: Lock state

**SHRRD**

The lock state is shared for read.

**SHRNUP**

The lock state is shared, no update.

**SHRUPD**

The lock state is shared for update.

**EXCLRD**

The lock state is exclusive, allow read.

**EXCL**

The lock state is exclusive, no read.

Element 4: Member, if data base file

**Note:** The following values can only be specified if the object type is a database file.

**FIRST**

The first member of the database file is deallocated.

*name* Specify the name of the member to be allocated. If the specified file is a logical file, the physical file members associated with the members of the logical file are also deallocated.

An explanation of how to specify multiple locks on an object, locking device descriptions, or the type objects that can be allocated, is in the CL information in the iSeries Information Center at http://www.ibm.com/eserver/iseries/infocenter.

If a database file member is deallocated, the first occurrence of the file in the library list is found and that file is searched for the specified member. If a file of the same name farther down on the library list contains the member, but the first file does not, the member is not found. If the member name is not specified for a database file, the member name defaults to *FIRST and the member that was created first in the file is deallocated.
<table>
<thead>
<tr>
<th>Object Type</th>
<th>Object Type Definition</th>
<th>Lock States</th>
</tr>
</thead>
<tbody>
<tr>
<td>*AUTL</td>
<td>Authorization List</td>
<td>+EXCL +EXCLRD +SHRUPD +SHRNUP +SHRRD</td>
</tr>
<tr>
<td>*BNDDIR</td>
<td>Binding directory</td>
<td>x x x x x</td>
</tr>
<tr>
<td>*CLD</td>
<td>C Locale description</td>
<td>x x x x x</td>
</tr>
<tr>
<td>*CRQD</td>
<td>Change request</td>
<td>x x x x x</td>
</tr>
<tr>
<td>*CSI</td>
<td>Communications side</td>
<td>x x x x x</td>
</tr>
<tr>
<td>*DEVD</td>
<td>Device Description</td>
<td>x x x</td>
</tr>
<tr>
<td>*DTAARA</td>
<td>Data area</td>
<td>x x x x x</td>
</tr>
<tr>
<td>*DTADCT</td>
<td>Data dictionary</td>
<td>x x x x</td>
</tr>
<tr>
<td>*DTAQ</td>
<td>Data queue</td>
<td>x x x x</td>
</tr>
<tr>
<td>*FCT</td>
<td>Forms control table</td>
<td>x x x x</td>
</tr>
<tr>
<td>*FILE</td>
<td>File</td>
<td>x x x x</td>
</tr>
<tr>
<td>*FNTRSC</td>
<td>Font resource</td>
<td>x x x x</td>
</tr>
<tr>
<td>*FNTBL</td>
<td>Font mapping table</td>
<td>x x x x</td>
</tr>
<tr>
<td>*FORMDF</td>
<td>Form definition</td>
<td>x x x x</td>
</tr>
<tr>
<td>*IMGCLG</td>
<td>Image catalog</td>
<td>x x x x</td>
</tr>
<tr>
<td>*IPXD</td>
<td>Internet packet</td>
<td>x x x x</td>
</tr>
<tr>
<td>*LIB</td>
<td>Library</td>
<td>x x x x</td>
</tr>
<tr>
<td>*LOCALE</td>
<td>Locale space object</td>
<td>x x x x</td>
</tr>
<tr>
<td>*MEDDFN</td>
<td>Media definition</td>
<td>x x x x</td>
</tr>
<tr>
<td>*MENU</td>
<td>Menu</td>
<td>x x x x</td>
</tr>
<tr>
<td>*MGTCOL</td>
<td>Management collection</td>
<td>x x x x</td>
</tr>
<tr>
<td>*MODULE</td>
<td>Module</td>
<td>x x</td>
</tr>
<tr>
<td>*MSGQ</td>
<td>Message queue</td>
<td>x x</td>
</tr>
<tr>
<td>*NODL</td>
<td>Node list</td>
<td>x x x x</td>
</tr>
<tr>
<td>*NTBD</td>
<td>NetBIOS description</td>
<td>x x x x</td>
</tr>
<tr>
<td>*NWSCFG</td>
<td>Network server</td>
<td>x x x x</td>
</tr>
<tr>
<td>*NWSDF</td>
<td>Network server description</td>
<td>x x x x</td>
</tr>
<tr>
<td>*OVL</td>
<td>Overlay</td>
<td>x x x x</td>
</tr>
<tr>
<td>*PAGDFN</td>
<td>Page definition</td>
<td>x x x x</td>
</tr>
<tr>
<td>*PAGSEG</td>
<td>Page segment</td>
<td>x x x x</td>
</tr>
<tr>
<td>*PDFMAP</td>
<td>PDF Map</td>
<td>x x x x</td>
</tr>
<tr>
<td>*PDG</td>
<td>Print descriptor group</td>
<td>x x x x</td>
</tr>
<tr>
<td>*PGM</td>
<td>Program</td>
<td>x x</td>
</tr>
<tr>
<td>*PNLGRP</td>
<td>Panel group</td>
<td>x x x x</td>
</tr>
<tr>
<td>*PSFCFG</td>
<td>Print service facility</td>
<td>x x x x</td>
</tr>
<tr>
<td>*QMFORM</td>
<td>Query management form</td>
<td>x x x x x</td>
</tr>
<tr>
<td>*QRMDR</td>
<td>Query management query</td>
<td>x x x x x</td>
</tr>
<tr>
<td>*QRDFN</td>
<td>Query definition</td>
<td>x x x x x</td>
</tr>
<tr>
<td>*S36</td>
<td>S/36 machine description</td>
<td>x x x</td>
</tr>
<tr>
<td>*SBD</td>
<td>Subsystem description</td>
<td>x</td>
</tr>
<tr>
<td>*SCHIDX</td>
<td>Search index</td>
<td>x x x x</td>
</tr>
<tr>
<td>*SQLPKG</td>
<td>Structured Query</td>
<td>x x x x</td>
</tr>
<tr>
<td>*SRVPGM</td>
<td>Service program</td>
<td>x x x x</td>
</tr>
<tr>
<td>*SSND</td>
<td>Session description</td>
<td>x x x x</td>
</tr>
<tr>
<td>*TIMZON</td>
<td>Time zone description</td>
<td>x x x x</td>
</tr>
<tr>
<td>*USRIDX</td>
<td>User index</td>
<td>x x x x</td>
</tr>
<tr>
<td>*USRQ</td>
<td>User queue</td>
<td>x x x x</td>
</tr>
<tr>
<td>*USRSPC</td>
<td>User space</td>
<td>x x x x</td>
</tr>
<tr>
<td>*VLDL</td>
<td>Validation list object</td>
<td>x x x x</td>
</tr>
<tr>
<td>*WSCST</td>
<td>Workstation</td>
<td>x x x x</td>
</tr>
</tbody>
</table>

'x' indicates the lock state is allowed for the object type.

Lock scope (SCOPE)

Specify the scope for this lock request.
The lock is scoped to the job.

The lock is scoped to the lock space attached to the current thread. If no lock space is attached, the lock is scoped to the job.

The lock is scoped to the thread.

All object types supported by the OBJ parameter support job-scoped locks. All object types supported by the OBJ parameter support lock-space-scoped locks. For DDM objects with a lock-space-scope, the lock on the remote system is scoped to the job. To determine if an object type supports thread-scoped locks refer to the figure Object Types that Support Thread Scope Locks.

The lock scope must match the scope of the locks currently allocated for the job or thread.

<table>
<thead>
<tr>
<th>Object Type</th>
<th>Thread Scope</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUTL</td>
<td></td>
</tr>
<tr>
<td>BNDDIR</td>
<td></td>
</tr>
<tr>
<td>CLD</td>
<td></td>
</tr>
<tr>
<td>CRQD</td>
<td></td>
</tr>
<tr>
<td>CSI</td>
<td></td>
</tr>
<tr>
<td>DEVD</td>
<td></td>
</tr>
<tr>
<td>DATAA</td>
<td>x</td>
</tr>
<tr>
<td>DATADCT</td>
<td>x</td>
</tr>
<tr>
<td>DTAQ</td>
<td></td>
</tr>
<tr>
<td>FCT</td>
<td>x</td>
</tr>
<tr>
<td>FILE</td>
<td>x</td>
</tr>
<tr>
<td>FNTSC</td>
<td></td>
</tr>
<tr>
<td>FNTTL</td>
<td></td>
</tr>
<tr>
<td>FORMDF</td>
<td></td>
</tr>
<tr>
<td>IMGCLG</td>
<td>x</td>
</tr>
<tr>
<td>IPXD</td>
<td>x</td>
</tr>
<tr>
<td>LIB</td>
<td>x</td>
</tr>
<tr>
<td>LOCALE</td>
<td></td>
</tr>
<tr>
<td>MEDI</td>
<td></td>
</tr>
<tr>
<td>MNU</td>
<td>Menu</td>
</tr>
<tr>
<td>MGTCOL</td>
<td>x</td>
</tr>
<tr>
<td>MODULE</td>
<td></td>
</tr>
<tr>
<td>MSGQ</td>
<td>x</td>
</tr>
<tr>
<td>NODEL</td>
<td></td>
</tr>
<tr>
<td>NTBD</td>
<td>x</td>
</tr>
<tr>
<td>NWSCFG</td>
<td></td>
</tr>
<tr>
<td>NWSD</td>
<td>x</td>
</tr>
<tr>
<td>OVL</td>
<td></td>
</tr>
<tr>
<td>PAGDFN</td>
<td></td>
</tr>
<tr>
<td>PAGSEG</td>
<td></td>
</tr>
<tr>
<td>PDFMAP</td>
<td>PDF Map</td>
</tr>
<tr>
<td>PGD</td>
<td></td>
</tr>
<tr>
<td>PGM</td>
<td>x</td>
</tr>
<tr>
<td>PNLRG</td>
<td>Panel group</td>
</tr>
<tr>
<td>PSFCFG</td>
<td></td>
</tr>
<tr>
<td>QFORM</td>
<td></td>
</tr>
<tr>
<td>QMQR</td>
<td></td>
</tr>
<tr>
<td>QRYDFN</td>
<td></td>
</tr>
<tr>
<td>S36</td>
<td>S/36 machine description x</td>
</tr>
<tr>
<td>SBSD</td>
<td>Subsystem description x</td>
</tr>
<tr>
<td>SCHIDX</td>
<td>Search index x</td>
</tr>
<tr>
<td>SQLPKG</td>
<td>Structured Query Language package</td>
</tr>
<tr>
<td>SRVPQM</td>
<td></td>
</tr>
<tr>
<td>SSND</td>
<td></td>
</tr>
<tr>
<td>TIZMIZ</td>
<td></td>
</tr>
<tr>
<td>USRIDX</td>
<td>User index x</td>
</tr>
<tr>
<td>USRQ</td>
<td>User queue x</td>
</tr>
<tr>
<td>USRSPC</td>
<td>User space x</td>
</tr>
<tr>
<td>VLDL</td>
<td>Validation list object x</td>
</tr>
<tr>
<td>WSCST</td>
<td>Workstation customizing object x</td>
</tr>
</tbody>
</table>

'x' indicates a thread-scoped lock is allowed for the object type.
Examples

Example 1: Deallocate a File for the Job
DLCOBJ OBJ((LIBB/FILEA *FILE *SHRRD))

This command releases the shared-for-read allocation of the first member of file FILEA in library LIBB that was held by the job.

Example 2: Deallocate a Data Area for the Thread
DLCOBJ OBJ((LIBY/DATAAREAX *DTAARA *SHRRD )) SCOPE(*THREAD)

This command releases the shared-for-read allocation of the data area DATAAREAX in library LIBY that was held by the thread.

Example 3: Deallocate File for Lock Space
DLCOBJ OBJ((LIBB/FILEA *FILE *EXCL MEMBERA)) SCOPE(*LCKSPC)

This command deallocates member MEMBERA of file FILEA in library LIBB from the lock space attached to the current thread.

Error messages

*ESCAPE Messages

CPF1005
Objects not deallocated.
Delete Cluster Admin Domain (DLTADMDMN)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Delete Cluster Administrative Domain (DLTADMDMN) command deletes the cluster resource group associated with a cluster administrative domain from all cluster nodes in the administrative domain. Cluster Resource Services must be active to use this command. The cluster resource group object is marked for deletion and is deleted on each active cluster node. The cluster resource group object will be deleted on other nodes in the cluster when they become active.

The Delete Cluster Resource Group (DLTCRG) command can be used to delete the cluster resource group object associated with a cluster administrative domain on a system that does not have Cluster Resource Services active.

Restrictions:
1. You must have input/output system configuration (*IOSYSCFG) special authority to run this command.
2. Cluster Resource Services must be active on the node processing the request.
3. The status of the cluster resource group being deleted must not be active.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLUSTER</td>
<td>Cluster</td>
<td>Name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td>ADMDMN</td>
<td>Cluster administrative domain</td>
<td>Name</td>
<td>Required, Positional 2</td>
</tr>
</tbody>
</table>

Cluster (CLUSTER)

Specifies the cluster containing the cluster administrative domain.

This is a required parameter.

*name* Specify the name of the cluster.

Cluster administrative domain (ADMDMN)

Specifies the cluster administrative domain that is to be deleted.

This is a required parameter.

*name* Specify the name of the cluster administrative domain that is to be deleted.
**Examples**

DLTADMMDN CLUSTER(MYCLUSTER) ADMMDN(MYDOMAIN)

This command deletes the cluster administrative domain named from the cluster MYCLUSTER.

**Error messages**

*ESCAPE Messages*

CPF1999

Errors occurred on command.
Delete Alert (DLTALR)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Delete Alert (DLTALR) command allows you to delete one or more alerts from the alert database.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>DLTOPT</td>
<td>Delete option</td>
<td>*ALL, *RCV, *LOCAL, *HELD</td>
<td>Optional, Positional 1</td>
</tr>
<tr>
<td>DAYS</td>
<td>Days</td>
<td>0-999, 30</td>
<td>Optional, Positional 2</td>
</tr>
<tr>
<td>ALRRSRC</td>
<td>Alert resource</td>
<td>Single values: *ALL, Other values (up to 50 repetitions): Name</td>
<td>Optional</td>
</tr>
<tr>
<td>ALRRSCYPE</td>
<td>Alert resource type</td>
<td>Single values: *ALL, Other values (up to 50 repetitions): Character value</td>
<td>Optional</td>
</tr>
<tr>
<td>ASNUSER</td>
<td>User assigned</td>
<td>Single values: *ALL, Other values (up to 50 repetitions): Character value, *NONE</td>
<td>Optional</td>
</tr>
<tr>
<td>GROUP</td>
<td>Group</td>
<td>Single values: *ALL, Other values (up to 50 repetitions): Name, *NONE, *DEFAULT</td>
<td>Optional</td>
</tr>
</tbody>
</table>

Delete option (DLTOPT)

Specifies which alerts are to be deleted.

*ALL    All alerts that meet the selection criteria in the remaining keywords are deleted. If the remaining keywords are defaulted, all of the alerts over 30 days old are deleted.

*RCV    Only alerts received from other systems are deleted. Selection criteria for the received alerts can be further specified in the remaining keywords. If the remaining keywords are defaulted, all of the received alerts over 30 days old are deleted.

*LOCAL  Only locally created alerts are deleted. Selection criteria for the local alerts can be further specified in the remaining keywords. If the remaining keywords are defaulted, all of the locally created alerts over 30 days old are deleted.

*HELD   All alerts that cannot be sent to the system’s focal point and are marked HELD are deleted. Selection criteria for the held alerts can be further specified in the remaining HELD keywords. If the remaining keywords are defaulted, all of the held alerts over 30 days old are deleted.
Note: There is a distinction between held alerts that are sent or forwarded by this system, and held alerts that are received by another system. DLTOPT(*HELD) deletes only held alerts that could not be sent or forwarded by this system (or are currently being held based on the ALRHLDCNT network attribute).

Days (DAYS)

Specifies that alerts older than this value are deleted. Alerts that are more recent than this value are not deleted. This value can be 0 or any number of days. Selection criteria can be further specified in the remaining keywords.

30 All alerts over 30 days old are deleted.
0-999 Specify the number of days.

Alert type (ALRTYPE)

Specifies which types of alerts are deleted. The alert type indicates the severity of the alert.

Single values

*ALL All types of alerts are deleted.

Other values (up to 5 repetitions)

*TEMP All alerts reporting a temporary problem are deleted.
*PERM All alerts reporting a permanent problem are deleted.
*PERF All alerts reporting a performance problem are deleted.
*IMPEND All alerts reporting an impending problem are deleted.
*UNKNOWN All alerts reporting a problem with unknown severity are deleted.

character-value Specify the code point for the alert type. The code point is specified by two (2) hexadecimal digits.

Alert resource (ALRRSC)

Specifies the name of resources that are reporting problems. Up to 50 alert resource names can be specified.

Single values

*ALL Alerts about all failing resources.

Other values (up to 50 repetitions)

name Specify the name of resources that are reporting problems.
Alert resource type (ALRRSCTYPE)

Specifies the type of resources that are reporting problems. A maximum of 50 resource types can be specified. Each resource name has a resource type associated with that resource. For example, resource types are diskette (DKT) or tape (TAP).

Single values

*ALL   Alerts for all resource types.

Other values (up to 50 repetitions)

character-value

Specify the resource type of alerts that are reporting problems associated with the assigned resource type.

User assigned (ASNUSER)

Specifies the user to which the alerts being deleted are assigned. This value is taken from the value on the ASNUSER parameter in the Add Alert Action Entry (ADDALRACNE) command.

Single values

*ALL   All alerts are deleted.

Other values (up to 50 repetitions)

*NONE   The alerts not assigned to a user are deleted.

character-value

Specify the name of the user to which the alerts being deleted are assigned.

Group (GROUP)

Specifies the group to which the alerts being deleted are assigned. This value is taken from the value on the GROUP parameter in the Add Alert Selection Entry (ADDALRSLTE) command.

Single values

*ALL   All alerts are deleted.

Other values (up to 50 repetitions)

*DEFAULT   The alerts assigned to the default group are deleted.

*NONE   The alerts not assigned to a group are deleted.

name   Specify the name of the group to which the alerts being deleted are assigned.
Examples

Example 1: Deleting Temporary and Permanent Alert Types

```
DLTALR   DLTOPT(*LOCAL)  DAYS(10)  ALRTYPE(*TEMP  *PERM)
          ALRRSCTYPE(DKT)
```

This command deletes temporary and permanent locally created alerts in the alert database. The alerts that are deleted are reporting problems about diskettes. Alerts more than 10 days old that match these criteria are deleted.

Example 2: Deleting Alerts Associated with Diskette Resources

```
DLTALR   DLTOPT(*RCV)   DAYS(0)   ALRRSCTYPE(DKT)
```

This command deletes received alerts associated with diskette resources.

Error messages

*ESCAPE Messages

CPF9807

One or more libraries in library list deleted.

CPF9808

Cannot allocate one or more libraries on library list.

CPF9812

File &1 in library &2 not found.

CPF9822

Not authorized to file &1 in library &2.

CPF9845

Error occurred while opening file &1.

CPF9846

Error while processing file &1 in library &2.

CPF9847

Error occurred while closing file &1 in library &2.
Delete Alert Table (DLTALRTBL)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Delete Alert Table (DLTALRTBL) command deletes an alert table from the specified library. More information on deleting alert tables is in the Alerts Support book, SC41-5413.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>ALRTBL</td>
<td>Alert table</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Alert table</td>
<td>Generic name, name</td>
<td></td>
</tr>
</tbody>
</table>

Alert table (ALRTBL)

Specifies the qualified name of the alert table being deleted.

The possible values are:

*alert-table-name*

Specify the name of the alert table being deleted.

*generic*-alert-table-name

Specify the generic name of the alert table being deleted. A generic name is a character string that contains one or more characters followed by an asterisk (*). If a generic name is specified for the ALRTBL parameter, all alert tables that have names with the same prefix as the generic alert table name are deleted.

The possible library values are:

*LIBL*  All libraries in the library list for the current thread are searched until the first match is found.

*USRLIBL*  If a current library entry exists in the library list for the current thread, the current library and the libraries in the user portion of the library list are searched. If there is no current library entry, only the libraries in the user portion of the library list are searched.

*CURLIB*  The current library list is searched to locate the alert table. If no library is specified as the current library for the job, then the QGPL library is used.

*ALL*  All libraries in the system, including QSYS, are searched.

*ALLUSR*  All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:
Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered user libraries and are also searched:

- QDSNX
- QGCL
- QGPL
- QGPL3B
- QMTC
- QMPDATA
- QMQMDATA
- QMQPROC
- QMPDATA
- QRCL

1. ‘xxxxx’ is the number of a primary auxiliary storage pool (ASP).
2. A different library name, in the format QUSRVxRxMx, can be created by the user for each previous release supported by IBM to contain any user commands to be compiled in a CL program for the previous release. For the QUSRVxRxMx user library, VxRxMx is the version, release, and modification level of a previous release that IBM continues to support.

library-name
Specify the library where the alert table is located.

Examples
DLTALRTBL ALRTBL(ALRTBLLIB/ALRTBLNBR1)

This command deletes alert table ALRTBLNBR1 from library ALRTBLLIB.

Error messages
*ESCAPE Messages
CPF2105
Object &1 in &2 type *&3 not found.
CPF2110
Library &1 not found.
CPF2113
Cannot allocate library &1.
CPF2114
Cannot allocate object &1 in &2 type *&3.
CPF2176
Library &1 damaged.
CPF2182
Not authorized to library &1.
CPF2189
Not authorized to object &1 in &2 type *&3.
Delete APAR Data (DLTAPARDTA)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Delete APAR Data (DLTAPARDTA) command deletes the Authorized Problem Analysis Report (APAR) library and any data created by the Restore APAR Data (RSTAPARDTA) command. This command also updates the problem log entry with new information and removes the referral to the APAR library.

Restrictions:
- The following user profiles have authority to this command:
  - QPGMR
  - QSYSOPR
  - QSRVBAS
  - QSRV

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>PRBID</td>
<td>Problem identifier</td>
<td>Character value</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td>ORIGIN</td>
<td>Origin</td>
<td>Element list</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Element 1: Network identifier</td>
<td>Communications name, *NETATR</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Control point name</td>
<td>Communications name, *NETATR</td>
<td></td>
</tr>
</tbody>
</table>

Problem identifier (PRBID)

Specifies the identifier (ID) of the problem for which APAR data is to be deleted.

This is a required parameter.

character-value

Specify the problem identifier to be used.

Origin (ORIGIN)

Specifies the network ID and the control point where the problem occurred.

Element 1: Network identifier
The problem originated on a system with the same local network ID as the one defined on the network attributes of this system.

Specify the network ID of the system where the problem originated.

Element 2: Control point name

The problem originated on a system with the same control point name as the one defined on the network attributes of this system.

Specify the control point of the system where the problem originated.

Examples

DLTAPAR DTA PRBID(9202448748)

This command deletes an APAR library and the APAR data for the problem ID 9202448748.

Error messages

*ESCAPE Messages

CPF2182
Not authorized to library &1.

CPF39FA
Problem &1 &2 &3 not found

CPF39FE
No APAR data associated with problem &1

CPF39F2
Cannot allocate library &1

CPF39F5
Query of problem &1 failed

CPF39F9
Problem &1 &2 &3 in use
Delete Authority Holder (DLTAUTHLR)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Delete Authority Holder (DLTAUTHLR) command allows a user to delete an authority holder that secures an object of type *FILE.

Restrictions:
- The authority holder can be deleted by a user who has all object (*ALLOBJ) special authority or all (*ALL) authority to the object it secures.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>OBJ</td>
<td>Object</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Object</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name</td>
<td></td>
</tr>
</tbody>
</table>

Object (OBJ)

Specifies the authorization holder object to be deleted.

This is a required parameter.

Qualifier 1: Object

name Specify the name of the authorization holder object.

Qualifier 2: Library

name Specify the name of the library where the authorization holder is located.

Examples

DLTAUTHLR OBJ(QGPL/FIL1)

This command deletes the authority holder for FIL1 in the QGPL library.
Error messages

*ESCAPE Messages

CPC2213
Authority holder deleted.

CPF22B1
Authority holder does not exist.

CPF22B2
Not authorized to create or delete authority holder.

CPF9803
Cannot allocate object &2 in library &3.
Delete Authorization List (DLTAUTL)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Delete Authorization List (DLTAUTL) command allows a user to delete an authorization list. Authorization lists cannot be deleted if they are being used to secure an object. The user deleting an authorization list must have all object (*ALLOBJ) special authority or be the owner of the authorization list.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>AUTL</td>
<td>Authorization list</td>
<td>Generic name, name</td>
<td>Required, Positional 1</td>
</tr>
</tbody>
</table>

Authorization list (AUTL)

Specifies the authorization lists to be deleted.

This is a required parameter.

generic-name

Specify the generic name of the authorization lists to be deleted. If a generic authorization list name is specified, then all authorization lists that have names with the same prefix as the generic authorization list name and that the user has proper authority for are deleted.

A generic name is a character string of one or more characters followed by an asterisk (*); for example ABC*. The asterisk substitutes for any valid characters. A generic name specifies all objects with names that begin with the generic prefix for which the user has authority. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete object name.

name

Specify the name of the authorization list to be deleted.

Examples

Example 1: Deleting an Authorization List

DLTAUTL AUTL(PROGMR)

This commands deletes the PROGMR authorization list.

Example 2: Deleting Generic Authorization Lists

DLTAUTL AUTL(FR*)
This commands deletes all authorization lists starting with the letters FR.

### Error messages

**ESCAPE Messages**

CPF2105  
Object &1 in &2 type *&3 not found.

CPF2110  
Library &1 not found.

CPF2113  
Cannot allocate library &1.

CPF2114  
Cannot allocate object &1 in &2 type *&3.

CPF2116  
DATA(*YES) specified and *ALL or *FILE not in OBJTYPE list.

CPF2117  
&4 objects type *&3 deleted. &5 objects not deleted.

CPF2125  
No objects deleted.

CPF2160  
Object type *&1 not eligible for requested function.

CPF2176  
Library &1 damaged.

CPF2182  
Not authorized to library &1.

CPF2189  
Not authorized to object &1 in &2 type *&3.

CPF2279  
Authorization list &1 cannot be deleted.

CPF2289  
Unable to allocate authorization list &1.

CPF5702  
File either not DDM file or not found.

CPF9801  
Object &2 in library &3 not found.
Delete Binding Directory (DLTBNDDIR)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Delete Binding Directory (DLTBNDDIR) command deletes a binding directory.

Restrictions
• You must have execute (*EXECUTE) authority to the library from which the binding directory is to be deleted and object existence (*OBJEXIST) authority to the binding directory.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>BNDDIR</td>
<td>Binding directory</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Binding directory</td>
<td>Generic name, name</td>
<td></td>
</tr>
</tbody>
</table>

Binding directory (BNDDIR)

Specifies a binding directory or a group of binding directories to be deleted.

This is a required parameter.

Qualifier 1: Binding directory

generic-name

Specify the generic name of the binding directories to be deleted. A generic name is a character string of one or more characters followed by an asterisk (*); for example, ABC*. If a generic name is specified, then all binding directories with names that begin with the generic name, and for which the user has authority, are deleted. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete binding directory name.

name

Specify the name of the binding directory to be deleted.

Qualifier 2: Library

*LIBL All libraries in the library list for the current thread are searched until the first match is found.

*CURLIB

The current library for the job is searched. If no library is specified as the current library for the job, the QGPL library is used.

*USRLIBL

Only the libraries in the user portion of the job’s library list are searched.

*ALL All libraries in the system, including QSYS, are searched.
*ALLUSR
All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

#CGULIB  #DSULIB  #SEULIB
#COBLIB  #RPGLIB
#DFULIB  #SDALIB

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered user libraries and are also searched:

QODSNX  QRLxxxxxx  QUSRJJS  QUSRVxRxMx
OGPL    QSRVAGT   QUSRINFSKR
QGPL3B  QSYS2     QUSRNOTES
QMTC    QSYS2xxxxx QUSROND
QMTC2   Q36F      QUSRPOSGS
QMPPDATA QUSR3B    QUSRPOSSA
QMMDATA QUSRADSM  QUSRPYMSVR
QMMPROC QUSRBM     QUSRADARS
QPFRDATA QUSRDIRCL QUSRYS
QRCL    QUSRDIRDB  QUSRVI

1. ‘xxxxx’ is the number of a primary auxiliary storage pool (ASP).
2. A different library name, in the format QUSRVxRxMx, can be created by the user for each previous release supported by IBM to contain any user commands to be compiled in a CL program for the previous release. For the QUSRVxRxMx user library, VxRxMx is the version, release, and modification level of a previous release that IBM continues to support.

Examples
DLTBNDDIR  BNDDIR(DISPLAYS)

This command deletes the binding directory named DISPLAYS using the job library list to locate the binding directory.

Error messages
*ESCAPE Messages

CPF2105
Object &1 in &2 type *&3 not found.

CPF2110
Library &1 not found.

CPF2113
Cannot allocate library &1.

CPF2114
Cannot allocate object &1 in &2 type *&3.

CPF2117
&4 objects type *&3 deleted. &5 objects not deleted.

CPF2125
No objects deleted.

CPF2160
Object type *&1 not eligible for requested function.
CPF2176
Library &1 damaged.

CPF2182
Not authorized to library &1.

CPF2189
Not authorized to object &1 in &2 type *&3.

CPFA030
Object already in use.

CPFE007
Error occurred processing command.

CPF9803
Cannot allocate object &2 in library &3.
The Delete Configuration List (DLTCFGL) command deletes the specified configuration list.

### Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>CFGL</td>
<td>Configuration list</td>
<td>Generic name, name</td>
<td>Required, Positional 1</td>
</tr>
</tbody>
</table>

### Configuration list (CFGL)

Specifies the name of one or more configuration list to delete. A specific configuration list or a generic configuration list can be specified.

- **configuration-list-name**
  - Specify the name of the configuration list to delete.

- **generic*-configuration-list-name**
  - Specify the generic name of the configuration list to delete.

**Note:** A generic name is specified as a character string that contains one or more characters followed by an asterisk (*). If a generic name is specified, then all objects that have names with the same prefix as the generic object name are selected.

This is a required parameter.

### Examples

`DLTCFGL CFGL(CONFIG01)`

This command deletes the configuration list CONFIG01 from the system.

### Error messages

***ESCAPE Messages**

**CPF2105**

Object &1 in &2 type *&3 not found.
CPF2625
Not able to allocate object &1.
Delete C Locale Description (DLTCLD)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Delete C Locale Description command deletes the C/400* locale description you specify on the CLD parameter. You can delete more than one locale by specifying a generic name as a locale name. If you specify a specific name for the C locale description, only the first occurrence of the locale is deleted when the DLTCLD is processed. The system searches for the locale based on the library specified. If the locale does not exist or is not found in the specified library, the locale is not deleted.

Error messages for DLTCLD

None

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLD</td>
<td>Locale name</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td>Qualifier 1: Locale name</td>
<td>Generic name, name</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Locale name (CLD)

Specifies the name of the C locale description you want to delete. You must enter a specific locale name or generic locale name.

locale-name

Specifies the name of the C locale description you want to delete.

generic*-C-locale-description-name

Specifies a generic name for the C Locale Descriptions that you want to delete. A generic name can contain a character string with one or more asterisks (*). If a generic name is specified for the CLD parameter, then all C locale descriptions that have names with the same prefix are deleted. For example, if you enter the locale name MY*, all locales beginning with MY in the specified library, are deleted. See the Control Language Reference more information on the use of generic functions.

The possible library values are:

*LIBL Both the user and system portions of the library list are searched for the specified locale. If a specific C locale description name is specified (instead of a generic name), only the first locale found with that name is deleted.

*CURLIB The current library for the job is searched. If you have not specified a current library, the system will search QGPL for the specified locale.
*USRLIBL
The user portion of the library list is searched for the specified locale. If a specific C locale description name is specified (instead of a generic name), only the first locale found with that name is deleted.

*ALL
All of the libraries in the system, including QSYS, are searched for the specified locale. If a specific C locale description name is specified (instead of a generic name), only the first locale found with that name is deleted.

*ALLUSR
All non-system libraries, including all user-defined libraries and QGPL, are searched. This includes both libraries in your library list and those not specified there. All libraries beginning with the letter Q, other than QGPL, are not searched, as these are system libraries.

library-name
Enter the name of the library you want to search for the specified locale. This is the only library that is searched when the DLTCLD command is processed. You must have *USE authority for the specified library.

Examples
None

Error messages
None
Delete Class (DLTCLS)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Delete Class (DLTCLS) command deletes a class object or a group of class objects from the system. Any routing steps that are running that are using the class are not affected by its deletion. However, additional routing steps using this class cannot be started. If the deleted class is referred to in any existing routing entry, either the routing entry must be changed (to refer to a different class) or another class must be created with the same name. If a subsystem routing entry specifies a deleted class, the subsystem is unable to start any jobs using that routing entry.

Restrictions:
1. To use this command, you must have object existence (*OBJEXIST) authority for the class, and execute (*EXECUTE) authority for the library.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLS</td>
<td>Class</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Class</td>
<td>Generic name, name</td>
<td></td>
</tr>
</tbody>
</table>

Class (CLS)

Specifies the name and library of the class being deleted. A specific class or a generic class can be specified; either type can be Optionally qualified by a library name.

This is a required parameter.

Qualifier 1: Class

generic-name

Specify the generic name of the class being deleted. A generic name is a character string that contains one or more characters followed by an asterisk (*). If a generic name is specified, then all classes that have names with the same prefix as the generic class are deleted.

name

Specify the name of the class.

Qualifier 2: Library

*LIBL

All libraries in the thread’s library list are searched until a match is found. If a specific object name is specified (instead of a generic name), only the first object found to have that name is deleted.
*CURLIB
The current library for the thread is used to locate the object. If no library is specified as the current library for the thread, the QGPL library is used.

*USRLIBL
Only the libraries listed in the user portion of the library list are searched. If a specific object name is specified (instead of a generic name), only the first object found with that name is deleted.

*ALL All libraries in auxiliary storage pools (ASPs) that are currently part of the thread’s library name space will be searched. This includes the system ASP (ASP 1), all defined basic user ASps (ASPs 2-32), and, if the thread has an ASP group, the primary and secondary ASps in the thread’s ASP group. Only your own QTEMP library is searched. All objects matching the specified name and object type in all libraries in the thread’s name space are deleted.

*ALLUSR All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

#CGULIB  #DSULIB  #SEULIB
#CBLIB   #RPGLIB
#DULIB   #DALIB

Although the following Qxxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered user libraries and are also searched:

QDSNX  QRCxxxxxx  QUSRIJS  QUSRVxRxMx
QGPL   QSRVAGT  QUSRINF SKR
QGPL3B QSYS2     QUSRNOTES
QMTC   QSYS2xxxxx QUSROND
QMTC2  QS36F     QUSRPOS GS
QMPCDATA QUSER3B  QUSRPOSSA
QMPCDATA QUSRADS M QUSRPOSSH
QMPCPROC QUSRBRM  QUSRDRDARS
QFPRDATA QUSRDIRCL QUSR SYS
QRC    QUSRDIRDB QUSRVI

1. ‘xxxxx’ is the number of a primary auxiliary storage pool (ASP).
2. A different library name, in the format QUSRVxRxMx, can be created by the user for each previous release supported by IBM to contain any user commands to be compiled in a CL program for the previous release. For the QUSRVxRxMx user library, VxRxMx is the version, release, and modification level of a previous release that IBM continues to support.

name Specify the library where the class is located.

Examples
DLTCLS  CLS(CLASS1)
This command deletes the class named CLASS1 from the system.

Error messages

*ESCAPE Messages

CPF2105 Object &1 in &2 type &3 not found.
CPF2110
Library &1 not found.

CPF2113
Cannot allocate library &1.

CPF2114
Cannot allocate object &1 in &2 type *&3.

CPF2117
&4 objects type *&3 deleted. &5 objects not deleted.

CPF2160
Object type *&1 not eligible for requested function.

CPF2176
Library &1 damaged.

CPF2182
Not authorized to library &1.

CPF2189
Not authorized to object &1 in &2 type *&3.

CPF5702
File either not DDM file or not found.
Delete Cluster (DLTCLU)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Delete Cluster (DLTCLU) command deletes a cluster on all nodes currently in the cluster’s membership list. All cluster resource group objects and device domains associated with the cluster are also deleted. Cluster Resource Services is ended on each node in the cluster membership list, and the cluster is deleted.

If this command is initiated from a cluster node with a status of Active, all active cluster nodes will be removed from the cluster, and the cluster resource group objects associated with the cluster will be deleted. Cluster resource group objects on nodes with a status of Inactive or Failed will not be deleted. If this command is initiated from a cluster node with a status of Failed or Inactive, only that node is removed from the cluster and cluster resource group objects on that node are deleted.

Cluster resource group exit programs will be called with an action code of Delete (or Delete Command if Cluster Resource Services is not active on the node where the command is invoked).

This command may be called when the cluster is in a partitioned state. In this case, the delete operation will only be performed within the partition running the command.

A node which was a member of a device domain has internal information related to auxiliary storage pools such as disk unit numbers or virtual memory addresses. After a cluster is deleted, this internal information persists until the node is IPLed. If the cluster is deleted, the node must be IPLed before the node can become a member of another device domain.

Restrictions:
1. You must have input/output system configuration (*IOSYSCFG) special authority to run this command.
2. This command cannot be called from a cluster resource group exit program.
3. This command must be called from a node defined in the cluster membership list.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLUSTER</td>
<td>Cluster</td>
<td>Name</td>
<td>Required, Positional 1</td>
</tr>
</tbody>
</table>

Cluster (CLUSTER)

Specifies the cluster which is being deleted.

This is a required parameter.

*name* Specify the name of the cluster that is to be deleted.
Examples

DLTCLU   CLUSTER(MYCLUSTER)

This command deletes the cluster MYCLUSTER. All nodes that were in the membership list of MYCLUSTER are no longer members of any cluster.

Error messages

*ESCAPE Messages

CPF0001

   Error found on &1 command.
Delete Command (DLTCMD)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Delete Command (DLTCMD) command deletes a user-defined command (or group of commands) from the library where it is located. Only the command definition object is removed; the command definition source, the command processing program, and the validity checker are not affected.

Restriction:
• You must have object existence (*OBJEXIST) authority to the command to be deleted.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMD</td>
<td>Command</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Command</td>
<td>Generic name, name</td>
<td></td>
</tr>
</tbody>
</table>

Command (CMD)

Specifies the commands to be deleted. A specific command name or a generic command name can be specified; optionally qualified by a library name.

Qualifier 1: Command

name  Specify the name of the command to be deleted.

generic-name

   Specify the generic name of the commands to be deleted. A generic name is a character string that contains one or more characters followed by an asterisk (*). If a generic name is specified, then all commands that have names with the same prefix as the generic name are deleted.

Qualifier 2: Library

*LIBL  All libraries in the library list for the current thread are searched until the first match is found.

*CURLIB

   The current library for the job is searched. If no library is specified as the current library for the job, QGPL is used.

*USRLIBL

   If a current library entry exists in the library list for the current thread, the current library and the libraries in the user portion of the library list are searched. If there is no current library entry, only the libraries in the user portion of the library list are searched.

   If a specific command name is specified (instead of a generic name), only the first command found with that name is deleted.
*ALL All libraries in the system, including QSYS, are searched.

*ALLUSR
All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

#CGULIB #DSULIB #SEULIB
#COBLIB #RPGLIB
#DFULIB #SDLIB

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered user libraries and are also searched:

QDSNX QRLxxxxx QUSRJJS QUSRVxRxMx
QGPL QSRVAGT QUSRINF SKR
QGPL3B QSYS2 QUSRNOTES
QMGTC QSYS2xxxxx QUSROND
QMGTC2 QS36F QUSRPOS GS
QMPGDATA QUSER3B QUSRPOSSA
QMOMDATA QUSRADSM QUSR YMSVR
QMOMPROC QUSRBM QUSRBDARS
QOPRDATA QUSRDIRCL QUSRSYS
QRL QUSRDIRDB QUSRVI

1. ‘xxxxx’ is the number of a primary auxiliary storage pool (ASP).
2. A different library name, in the format QUSRVxRxMx, can be created by the user for each previous release supported by IBM to contain any user commands to be compiled in a CL program for the previous release. For the QUSRVxRxMx user library, VxRxMx is the version, release, and modification level of a previous release that IBM continues to support.

name Specify the name of the library to be searched.

Examples

DLTCDM CMD(LIB01/PAYROLL)

This command deletes the command named PAYROLL from library LIB01. Any private authorities to the command are removed from the user profiles of all authorized users.

Error messages

*ESCAPE Messages

CPF2105
Object &1 in &2 type *&3 not found.

CPF2110
Library &1 not found.

CPF2113
Cannot allocate library &1.

CPF2114
Cannot allocate object &1 in &2 type *&3.

CPF2117
&4 objects type *&3 deleted. &5 objects not deleted.
CPF2176
Library &1 damaged.

CPF2182
Not authorized to library &1.

CPF2189
Not authorized to object &1 in &2 type *&3.
Delete Communications Trace (DLTCMNTRC)

Where allowed to run: All environments (*ALL)
TheDelete Communications Trace (DLTCMNTRC) command deletes the communications trace for the specified line, a network interface description, or a network server description. The communications trace can be deleted after the trace has ended.

Restrictions:
- To use this command, you must have service (*SERVICE) special authority, or be authorized to the Service Trace function of Operating System through iSeries Navigator’s Application Administration support. The Change Function Usage (CHGFCNUSG) command, with a function ID of QIBM_SERVICE_TRACE, can also be used to change the list of users that are allowed to perform trace operations.
- The following user profiles have authority to this command:
  - QSECOFR
  - QSROV

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>CFGOBJ</td>
<td>Configuration object</td>
<td>Name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td>CFGTYPE</td>
<td>Type</td>
<td>*LIN, *NWI, *NWS</td>
<td>Required, Positional 2</td>
</tr>
</tbody>
</table>

Configuration object (CFGOBJ)

Specifies the configuration object for which collected trace information is to be deleted.

This is a required parameter.

name Specify the name of the configuration object for which trace information is to be deleted.

Type (CFGTYPE)

Specifies the type of configuration description that was traced.

This is a required parameter.

*LIN The type of configuration object is a line description.
*NWI The type of configuration object is a network interface description.
The type of configuration object is a network server description.

**Examples**

```
DLTCMNT RC CFGOBJ(*QESLINE) CFGTYPE(*NWI)
```

This command deletes the communications trace data for line description QESLINE.

**Error messages**

**ESCAPE Messages**

CPF2634  
Not authorized to object &1.

CPF39A7  
Trace storage not available in communications processor

CPF39A8  
Not authorized to communications trace service tool

CPF39A9  
Error occurred during communications trace function

CPF39B0  
No communications traces exist.

CPF39B1  
Trace &1 type &2 does not exist

CPF39B2  
Trace data for &1 type &2 cannot be deleted

CPF39B6  
Communications trace function cannot be performed

CPF98A2  
Not authorized to &1 command.
Delete Connection List (DLTCNNL)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Delete Connection List (DLTCNNL) command deletes a connection list.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNNL</td>
<td>Connection list</td>
<td>Generic name, name</td>
<td>Required, Positional 1</td>
</tr>
</tbody>
</table>

Connection list (CNNL)

Specifies the name of the connection list.

This is a required parameter.

*name*: Specify the name of the connection list to be deleted.

*generic-name*: Specify the generic name of the connection list to be deleted. A generic name is a character string that contains one or more characters followed by an asterisk (*). If a generic name is specified for this parameter, all connection lists that have names with the same prefix as the generic connection list name are deleted.

Examples

Example 1: Deleting a Specified Connection List
DLTCNNL  CNNL(CHICAGO)

This command deletes the connection list named CHICAGO. Any entries in connection list CHICAGO are also deleted.

Example 2: Deleting Multiple Connection Lists
DLTCNNL  CNNL(CHI*)

This command deletes all connection lists whose name begins with CHI. All entries that were listed in the connection lists are also deleted.
Error messages

*ESCAPE Messages

CPF2625
   Not able to allocate object &1.

CPF2634
   Not authorized to object &1.

CPF266C
   Connection list &1 not found.

CPF267D
   Connection list &1 not deleted.

CPF268F
   Connection list &1 not deleted.
Delete Class-of-Service Desc (DLTCOSD)

Where allowed to run: All environments (*ALL)
Threadsafes: No

The Delete Class-of-Service Description (DLTCOSD) command deletes the specified class-of-service description.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>COSD</td>
<td>Class-of-service description</td>
<td>Qualifier list</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Class-of-service description</td>
<td>Generic name, name</td>
<td></td>
</tr>
</tbody>
</table>

Class-of-service description (COSD)

Specifies the names of one or more class-of-service description to delete. A specific class-of-service description or a generic class-of-service description can be specified.

class-of-service-description-name

Specify the name of the class-of-service description to delete.

generic*-class-of-service-description-name

Specify the generic name of the class-of-service description to delete.

Note: A generic name is specified as a character string that contains one or more characters followed by an asterisk (*). If a generic name is specified, then all objects that have names with the same prefix as the generic object name are selected.

Examples

DLTCOSD COSD(COS01)

This command deletes the class-of-service description named COS01 from the system.

Error messages

*ESCAPE Messages

CPF2105

Object &1 in &2 type *&3 not found.
CPF2634
Not authorized to object &l.
Delete Cluster Resource Group (DLTCRG)

Where allowed to run: All environments (*ALL)
Threatsafe: No

The Delete Cluster Resource Group (DLTCRG) command deletes a cluster resource group (CRG) on the local system only. Deleting a local cluster resource group requires the Cluster Resource Services to be inactive.

This command will cause the cluster resource group exit program to be invoked with an action code of Delete Command on the local system only.

If the CRG object exists on other nodes in the cluster, it is the owners responsibility to remove the node ID of the local system from the CRG on those nodes. This can be done using the Remove Node From Recovery Domain (RMVCRGNODE) command.

The cluster resource group object exit program will run under the user profile specified in the cluster resource group. If the exit program does not exist, a message is logged and the CRG object deletion will continue. This command will never invoke the cluster resource group exit program with an action code of Undo.

Restrictions:
• You must have input/output system configuration (*IOSYSCFG) special authority to run this command.
• You must have object existence (*OBJEXIST) and use (*USE) authorities to the cluster resource group being deleted.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>CRG</td>
<td>Cluster resource group</td>
<td>Name</td>
<td>Required, Positional 1</td>
</tr>
</tbody>
</table>

Cluster resource group (CRG)

Specifies the cluster resource group that is to be deleted.

This is a required parameter.

name Specify the name of the cluster resource group that is to be deleted.
Examples

DLTCRG CRG(CRGTEST)

This command deletes the cluster resource group named CRGTEST from the local system.

Error messages

*ESCAPE Messages

CPFBB39
   Current user does not have IOSYSCFG special authority.

CPFBB41
   Cluster resource group &1 exit program ended abnormally.

CPFBB46
   Cluster Resource Services internal error.

CPFBB53
   Cluster Resource Services is active. Request cannot be processed.

CPF2105
   Object &1 in &2 type *&3 not found.

CPF2110
   Library &1 not found.

CPF2113
   Cannot allocate library &1.

CPF2114
   Cannot allocate object &1 in &2 type *&3.

CPF2125
   No objects deleted.

CPF2182
   Not authorized to library &1.

CPF2189
   Not authorized to object &1 in &2 type *&3.

CPF9801
   Object &2 in library &3 not found.

CPF9802
   Not authorized to object &2 in &3.

CPF9803
   Cannot allocate object &2 in library &3.

CPF9804
   Object &2 in library &3 damaged.

CPF9810
   Library &1 not found.

CPF9820
   Not authorized to use library &1.

CPF9872
   Program or service program &1 in library &2 ended. Reason code &3.
Delete CRG Cluster (DLTCRGCLU)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Delete Cluster Resource Group from Cluster (DLTCRGCLU) command deletes a cluster resource group from all cluster nodes in the recovery domain. Cluster Resource Services must be active to use this command. The cluster resource group object is marked for deletion and is deleted on each active cluster node. The cluster resource group object will be deleted on other nodes in the cluster when they become active.

Deleting a device cluster resource group will not change the ownership of devices. The devices remain on whatever node owns them at the time of the delete.

If Cluster Resource Services configured the takeover IP address for an application cluster resource group and the IP interface is not active, the takeover IP address will be removed. If Cluster Resource Services finds that the takeover IP address is active, the command will fail.

If an exit program is specified for the cluster resource group, the cluster resource group exit program is called on each active node in the recovery domain with an action code of Verification Phase and action code dependent data of Delete. The cluster resource group status is set to Delete Pending. The cluster resource group exit program is called with an action code of Delete if the verification phase is successful. The cluster resource group will not be deleted if the verification phase fails. This command will not call the cluster resource group exit program with an action code of Undo when the verification phase fails.

The Delete Cluster Resource Group (DLTCRG) command can be used to delete a cluster resource group object on a system that does not have Cluster Resource Services active.

Restrictions:
1. You must have input/output system configuration (*IOSYSCFG) special authority to run this command.
2. Cluster Resource Services must be active on the node processing the request.
3. The status of the cluster resource group being deleted must not be active.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>CLUSTER</td>
<td>Cluster</td>
<td>Name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td>CRG</td>
<td>Cluster resource group</td>
<td>Name</td>
<td>Required, Positional 2</td>
</tr>
</tbody>
</table>
Cluster (CLUSTER)
Specifies the cluster containing the cluster resource group.

This is a required parameter.

`name` Specify the name of the cluster.

Cluster resource group (CRG)
Specifies the cluster resource group that is to be deleted.

This is a required parameter.

`name` Specify the name of the cluster resource group that is to be deleted.

Examples

```
DLTCRGCLU CLUSTER(MYCLUSTER) CRG(MYCRG)
```

This command deletes the cluster resource group named MYCRG from the cluster named MYCLUSTER.

Error messages

*ESCAPE Messages

`CPF1999`

Errors occurred on command.
Delete CRQ Description (DLTCRQD)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Delete Change Request Description (DLTCRQD) command deletes one or more change request descriptions.

Restriction: You must have object existence authority in order to delete the change request description.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>CRQD</td>
<td>Change request description</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Change request description</td>
<td>Generic name, name</td>
<td></td>
</tr>
</tbody>
</table>

Change request description (CRQD)

Specifies the name and the library of the change request description to delete.

The possible values are:

Qualifier 2: Library

*LIBL   All libraries in the library list for the current thread are searched until the first match is found.
*CURLIB The current library for the thread is searched. If no library is specified as the current library for the thread, the QGPL library is searched.
*USRLIBL If a current library entry exists in the library list for the current thread, the current library and the libraries in the user portion of the library list are searched. If there is no current library entry, only the libraries in the user portion of the library list are searched.
*ALLUSR All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:
    #CGULIB   #DSULIB   #SEULIB
    #COBLIB   #RPGLIB   #DFULIB  #SDALIB

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered user libraries and are also searched:
1. ‘xxxxx’ is the number of a primary auxiliary storage pool (ASP).

2. A different library name, in the format QUSRVxRxMx, can be created by the user for each previous release supported by IBM to contain any user commands to be compiled in a CL program for the previous release. For the QUSRVxRxMx user library, VxRxMx is the version, release, and modification level of a previous release that IBM continues to support.

*ALL  All libraries in the system, including QSYS, are searched.

name  Specify the name of the library to be searched.

change-request-description-name  Specify the name of the change request description to be deleted.

generic*-change-request-description-name  Specify a generic change request description name. A generic name is a character string containing one or more characters followed by an asterisk (*). All change request descriptions with names that match or that begin with the specified string are deleted.

Examples

Example 1: Deleting a Change Request Description from Your Own Library

DLTCRQD  CRQD(MYLIB/CHG001)

This command deletes a change request description in MYLIB with the name CHG001.

Example 2: Deleting Change Request Descriptions Using a Generic Name

DLTCRQD  CRQD(*LIBL/CHG*)

This command deletes all the change request descriptions in the library list whose names start with CHG.

Example 3: Deleting a Change Request Description from the Current Library

DLTCRQD  CRQD(*CURLIB/CHG456)

This command deletes the change request description named CHG456 in the current library.

Error messages

None
Delete Comm Side Information (DLTCSI)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Delete Communications Side Information (DLTCSI) command deletes a side information object from the library or libraries specified.

### Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>CSI</td>
<td>Side information</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Side information</td>
<td>Generic name, name</td>
<td></td>
</tr>
</tbody>
</table>

### Side information (CSI)

Specifies the name of the side information object to be deleted.

This is a required parameter.

The possible library values are:

*LIBL    All libraries in the library list for the current thread are searched until the first match is found.

*CURLIB  The current library is searched.

*USRLIBL If a current library entry exists in the library list for the current thread, the current library and the libraries in the user portion of the library list are searched. If there is no current library entry, only the libraries in the user portion of the library list are searched.

If a specific symbolic destination name is specified (instead of a generic name), only the first symbolic destination name found with that name is deleted.

*ALLUSR  All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

- #CGULIB
- #DSULIB
- #SEULIB
- #COBLIB
- #RPGLIB
- #DFULIB
- #SDALIB

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered user libraries and are also searched:

- QDSNX
- QRCXLxxxx
- QUSRJJS
- QUSRvRxMx
- QGPL
- QSRVAGT
- QUSRINFSKR
- QGPL3B
- QSYS2
- QUSRNOTES

© Copyright IBM Corp. 1998, 2006
1. ‘xxxx’ is the number of a primary auxiliary storage pool (ASP).
2. A different library name, in the format QUSRVxRxMx, can be created by the user for each previous release supported by IBM to contain any user commands to be compiled in a CL program for the previous release. For the QUSRVxRxMx user library, VxRxMx is the version, release, and modification level of a previous release that IBM continues to support.

*ALL All libraries in the system, including QSYS, are searched.

library-name
Specify the name of the library from which the side information object is to be deleted.

side-information-name
Specify the name of the object that contains the desired side information object.

generic*-side-information-name
Specify the generic name of the object that contains the desired side information object. A generic name is a character string that contains one or more characters followed by an asterisk (*).

---

**Examples**

DLTCSI CSI(QGPL/SIDEOBJ)

This command deletes the communications side information object named SIDEOBJ from library QGPL.

---

**Error messages**

None
Delete Controller Description (DLTCTLD)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Delete Controller Description (DLTCTLD) command deletes the specified controller description. A controller description must be varied offline before this command is issued to delete it.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>CTLD</td>
<td>Controller description</td>
<td>Generic name, name</td>
<td>Required, Positional 1</td>
</tr>
</tbody>
</table>

Controller description (CTLD)

Specifies the name of the controller description.

controller-description-NAME
Specify the NAME of the controller description to delete.

generic*-controller-description-NAME
Specify the generic NAME of the controller description to delete.

Note: A generic name is specified as a character string that contains one or more characters followed by an asterisk (*). If a generic name is specified, then all objects that have names with the same prefix as the generic object name are selected.

Examples

DLTCTLD  CTLD(CONTROL01)

This command deletes the controller description named CONTROL01 from the system. If the controller description being deleted has any device descriptions associated with it, they are detached and a message containing their names is sent to the system operator.

Error messages

*ESCAPE Messages

CPF2105
Object &1 in &2 type *&3 not found.
CPF2615
Controller &1 must be varied off for this operation.

CPF2634
Not authorized to object &1.

CPF2636
Command not processed. Controller &1 in use.

CPF2697
The request did not complete in the time allotted.

CPF2717
Controller description &1 not deleted.

CPF2782
Message &1 not monitored.
Delete Device Description (DLTDEVD)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Delete Device Description (DLTDEVD) command deletes the specified device description. The device description must be varied offline before this command is issued to delete it.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEVD</td>
<td>Device description</td>
<td>Generic name, name</td>
<td>Required, Positional 1</td>
</tr>
</tbody>
</table>

Device description (DEVD)

Specifies the names of one or more of the device description to delete. A specific device description or a generic device description can be specified.

device-description-name
    Specify the name of the device description to delete.

generic*-device-description-name
    Specify the generic name of the device description to delete.

Note: A generic name is specified as a character string that contains one or more characters followed by an asterisk (*). If a generic name is specified, then all objects that have names with the same prefix as the generic object name are selected.

Examples

DLTDEVD DEVD(DSPC01)

This command deletes the device description of the device named DSPC01 from the system.

Error messages

*ESCAPE Messages

CPF2105
    Object &1 in &2 type *&3 not found.
CPF2114
Cannot allocate object &1 in &2 type *&3.

CPF2117
&4 objects type *&3 deleted. &5 objects not deleted.

CPF2189
Not authorized to object &1 in &2 type *&3.

CPF2605
Not able to allocate &1.

CPF2615
Controller &1 must be varied off for this operation.

CPF2616
Device Description &1 not deleted. Device in use.

CPF2634
Not authorized to object &1.

CPF2646
Delete request failed.

CPF2648
Delete request not allowed for device &1.

CPF2668
Object description not deleted.

CPF268E
Device description &1 cannot be deleted or renamed.

CPF2697
The request did not complete in the time allotted.

CPF2782
Message &1 not monitored.
Delete Device Media Library (DLTDEVMLB)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Delete Device Media Library (DLTDEVMLB) command deletes a library device name from the system that had previously been created with the Create Device Media Library (CRTDEVMLB) command.

### Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>DEVD</td>
<td>Device description</td>
<td>Name</td>
<td>Required, Positional 1</td>
</tr>
</tbody>
</table>

### Device description (DEVD)

Specifies the library device to be deleted from the system.

This is a required parameter.

*name* Specify the name of the library device.

### Examples

DLTDEVMLB DEVD(TAPMLB01)

This command deletes the tape media library device description TAPMLB01.

### Error messages

None
IBM Systems - iSeries: i5/OS Commands Starting with CRTMSGF (Create Message File)
Delete DFU Program (DLTDFUPGM)

Where allowed to run: Interactive environments (*INTERACT
*IPGM *IREXX *EXEC)
Threadsafe: No

The Delete DFU Program (DLTDFUPGM) command deletes a DFU program from a library.

Error messages for DLTDFUPGM

None

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>DFUPGM</td>
<td>DFU program</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: DFU program</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, *LIBL, *CURLIB</td>
<td></td>
</tr>
</tbody>
</table>

DFU program (DFUPGM)

Specifies the qualified name of the DFU program that will be deleted.

*LIBL   DFU will use your library list to search for a specified program.
*CURLIB Type *CURLIB to use your current library. If no current library entry exists in the library list, QGPL is used. If you do not specify a library name, *LIBL is used.

Examples

None

Error messages

None
IBM Systems - iSeries: i5/OS Commands Starting with CRTMSGF (Create Message File)
Delete Document Library Object (DLTDLO)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Delete Document Library Object (DLTDLO) command allows you to delete a document or a folder.

This command is used to delete documents and folders that are no longer needed from the system, which prevents system storage from being used for obsolete objects.

Note: When a document is deleted, a request is generated to remove its entry (if it exists) from the text index.

Restrictions:
- To delete a document or folder, you must have all (*ALL) authority for the document or folder, all object (*ALLOBJ), or security administrator (*SECADM) special authority.
- You must have *ALLOBJ or *SECADM special authority to delete all documents or folders or to specify an owner other than yourself.
- Once you press the Enter key, the specified document or folder is deleted; there is no prompting before the document or folder is deleted.
- If the document or folder is in use, the document or folder is not deleted.
- When more than one document or folder is specified for deletion but one of the documents or folders cannot be deleted because it is in use, or you are not authorized to it, a message is sent, and the function continues to delete those documents or folders that remain in the list.
- If a folder and all documents and folders within it are being deleted, any document or folder that cannot be deleted from the folder remains, but all others are deleted. A message is sent concerning those documents and folders that cannot be deleted.
- While using this command, you may encounter an error message indicating that internal objects are locked. Another user is using document library functions which cannot run at the same time as the DLTDLO command; therefore, retry this command in a few minutes.
- An ASP value of *ANY can be specified only when DLO(*ALL) FLR(*ANY) or DLO(*SEARCH) FLR(*ANY) is specified.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>DLO</td>
<td>Document library object</td>
<td>Single values: *SEARCH, *ALL, *SYSOBJNAM Other values (up to 300 repetitions): Character value</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td>FLR</td>
<td>In folder</td>
<td>Character value, *ANY, *NONE</td>
<td>Optional, Positional 2</td>
</tr>
<tr>
<td>Keyword</td>
<td>Description</td>
<td>Choices</td>
<td>Notes</td>
</tr>
<tr>
<td>---------</td>
<td>---------------------</td>
<td>------------------------------</td>
<td>-------</td>
</tr>
<tr>
<td>CRTDATE</td>
<td>Creation period</td>
<td>Element list</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Element 1: Starting time and date</td>
<td>Element list</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 1: Starting time</td>
<td>Time, *AVAIL</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Starting date</td>
<td>Date, *CURRENT, *BEGIN</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Ending time and date</td>
<td>Element list</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 1: Ending time</td>
<td>Time, *AVAIL</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Ending date</td>
<td>Date, *END</td>
<td></td>
</tr>
<tr>
<td>CHKEXP</td>
<td>Expiration date</td>
<td>Date, *NO, *CURRENT</td>
<td>Optional</td>
</tr>
<tr>
<td>DOCCLS</td>
<td>Document class</td>
<td>Character value, *ANY</td>
<td>Optional</td>
</tr>
<tr>
<td>OWNER</td>
<td>Owner profile</td>
<td>Name, *CURRENT, *ALL</td>
<td>Optional</td>
</tr>
<tr>
<td>SYSOBJNAM</td>
<td>System object name</td>
<td>Values (up to 300 repetitions): Name</td>
<td>Optional</td>
</tr>
<tr>
<td>CMDCHRID</td>
<td>Command character identifier</td>
<td>Single values: *SYSVAL, *DEVD</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Element 1: Graphic character set</td>
<td>Integer</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Code page</td>
<td>Integer</td>
<td></td>
</tr>
<tr>
<td>ASP</td>
<td>Auxiliary storage pool ID</td>
<td>1-32, *ANY</td>
<td>Optional</td>
</tr>
</tbody>
</table>

**Document library object (DLO)**

Specifies the documents or folders that are deleted. If DLO(*SEARCH) is not specified, CRTDATE, CHKEXP, DOCCLS, CMDCHRID, and OWNER cannot be specified.

This is a required parameter.

*ALL  All documents or folders currently filed in the folder specified on the **In folder (FLR)** parameter are deleted.

*SEARCH  All documents and folders that meet the specified search values are deleted. Search values are specified using the following parameters. When more than one parameter is specified, the "AND" relationship is used to combine them:

- **In folder (FLR)** parameter
- **Creation period (CRTDATE)** parameter
- **Expiration date (CHKEXP)** parameter
- **Document class (DOCCLS)** parameter
- **Owner profile (OWNER)** parameter

When *SEARCH is specified, the search is performed by examining the document details.

*SYSOBJNAM  The system object names for the documents or folders specified on the **System object name (SYSOBJNAM)** parameter are used.

**name**  Specify one or more document or folder names of specific documents or folders that are deleted. A maximum of 300 documents or folders can be specified.
Note: To delete a folder and all documents and folders within it, specify *ALL on the Document library object (DLO) parameter and the folder name using the In folder (FLR) parameter. Otherwise, if a folder name is specified on the Document library object (DLO) parameter, the folder must be empty to be deleted.

**In folder (FLR)**

Specifies the folder that contains the documents or folders. If the document or folder does not exist in a folder, *NONE is specified.

*NONE

The document or folder is not contained in a folder. The *SEARCH and *ALL values on the Document library object (DLO) parameter apply only to documents and folders not contained in folder.

*ANY

A system-wide search takes place. The *SEARCH and *ALL values on the Document library object (DLO) parameter apply to all documents and folders, including those which are not contained in any folder.

**name**
Specify the folder name that contains the documents or folders.

**Creation period (CRTDATE)**

Specifies that documents and folders created during the time period specified are deleted. If this parameter is specified, *SEARCH must be specified on the Document library object (DLO) parameter.

**Element 1: Starting time**

*AVAIL

Documents and folders created at any time are deleted.

time

Specify a time. Only those documents and folders created at or after the specified time on the starting date are deleted. The time can be specified with or without a time separator:

- Without a time separator, specify a string of 4 or 6 digits (hhmm or hhmmss) where hh = hours, mm = minutes, and ss = seconds.
- With a time separator, specify a string of 5 or 8 digits where the time separator specified for your job is used to separate the hours, minutes, and seconds. If you enter this command from the command line, the string must be enclosed in apostrophes. If a time separator other than the separator specified for your job is used, this command will fail.
- Specify 0 for the seconds value as the seconds are not used in the selection process.

When the start-time is used as a search value, the start-date must not be *BEGIN.

**Element 2: Starting date**

*CURRENT

Only those documents and folders created on the day you enter this command, and after the starting time (if specified), are deleted.

*BEGIN

Documents and folders created on any specified date are deleted. (The starting date and time is ignored.)

date

Specify a date. Only those documents and folders created on or after the date specified are deleted. The date must be specified in the job date format.
Element 3: Ending time

*AVAIL

Documents and folders created at any time are deleted.

time

Specify a time. Only those documents and folders created at or before the specified time on the ending date are deleted. See the start-time on this parameter for details about how time must be specified. When the end-time is used as search values, the end-date must not be *END.

Element 4: Ending date

*END

Documents and folders created on any date are deleted. The ending time is ignored when *END is specified.

date

Specify a date. Only documents and folders created on or before this date are deleted. The date must be specified in the job date format.

Expiration date (CHKEXP)

Specifies the expiration date of the documents being deleted. The expiration date is assigned to specify when a document is no longer needed. A document with an expiration date (as specified by the user on the Change Document Details display) earlier than the date specified is deleted. If this parameter is specified, *SEARCH must also be specified on the Document library object (DLO) parameter.

*NO

The expiration date for the document is ignored.

*CURRENT

Only documents with an expiration date earlier than today’s date are deleted.

date

Specify a date. Only documents with an expiration date earlier than the date specified are deleted.

Document class (DOCCLS)

Specifies the class of documents that are deleted. If this parameter is specified, *SEARCH must also be specified on the Document library object (DLO) parameter.

*ANY

Documents are deleted without regard to the document’s class.

class

Specify a character string, ranging from 1 through 16 characters in length. Only documents of the specified class are to be deleted. For comparison, the document class specified and the document class of all documents are converted to uppercase.

Owner profile (OWNER)

Specifies the owner of the documents and folders that are deleted. If this parameter is specified, *SEARCH must also be specified on the Document library object (DLO) parameter.

*CURRENT

Only documents or folders owned by you are deleted.

*ALL

Document and folders are deleted without regard to the object’s owner. The user must have all object (*ALLOBJ) or security administrator (*SECADM) special authority to specify *ALL.
name  Specify the name of a user. Only documents or folders owned by the specified user are deleted.
You must have *ALLOBJ or *SECADM special authority to specify an owner other than yourself.

System object name (SYSOBJNAM)
Specifies the system object names of the document library objects that are deleted. A maximum of 300
system objects can be specified. System object names can be specified only if *SYSOBJNAM is specified
on the Document library object (DLO) parameter.
name  Specify one or more system object names that are deleted.

Command character identifier (CMDCHRHD)
Specifies the character identifier (graphic character set and code page) for the data being entered as
command parameter values (applies to the Document class (DOCCLS) parameter. The character
identifier is related to the display device that was used to enter the command.

*SYSVAL
  The system determines the graphic character set and code page values for the command
parameters from the QCHRID system value.

*DEVD
  The system determines the graphic character set and code page values for the command
parameter from the display device description where this command is entered. This option is
valid only when specified from an interactive job. If this value is specified in an interactive CL
program or a batch job, an error message is sent.

graphic-character-set-code-page
  Specify the graphic character set and code page values that are used to create the command
parameter. Each value can be up to 3 digits in length.

Auxiliary storage pool ID (ASP)
Specifies the identifier (ID) of the auxiliary storage pool (ASP) of the document library object to be
deleted. A value other than *ANY can be specified on this parameter only if *ALL or *SEARCH is
specified on the DLO parameter and *ANY is specified on the FLR parameter.

*ANY  Document library objects on the system are deleted without regard to the object’s ASP.
1-32   Only the document library objects that reside in the specified ASP are to be deleted. All
document library objects in other ASPs are ignored. The value must designate an existing ASP
that contains document library objects. ASP 1 is the system ASP.

Note: This option is provided to aid in recovery from damage to a user ASP, particularly one that
has overflowed. The document library objects and their details are deleted from the damaged
ASP (after the necessary data recovery actions are taken for the ASP), thereby removing
overflowed objects and potentially damaged objects from the ASP prior to restoring from backup
media. More information on document library objects in ASPs is in the Backup and Recovery
book, SC41-5304.
Examples

Example 1: Deleting a Folder
DLTDLO DLO(ABC)

This command deletes folder ABC if it contains no documents or folders.

Example 2: Deleting All Documents and Folders Within a Folder
DLTDLO DLO(*ALL) FLR(ABC)

This command deletes all the documents and folders that the user is authorized to delete from folder ABC. If folder ABC turns out to be empty, then it is also deleted.

Example 3: Deleting All Documents in a Document Class, Using *SECADM Special Authority
DLTDLO DLO(*SEARCH) CRTDATE((AVAIL 080187) (AVAIL 083187)) DOCCLS(LETTERS) OWNER(*ALL)

This command deletes all documents in document class LETTERS that were filed in the system during August 1987. Only the security officer or a user with *SECADM special authority is allowed to specify OWNER(*ALL).

Example 4: Deleting Current User's Documents
DLTDLO DLO(*SEARCH) CRTDATE((AVAIL +BEGIN) (AVAIL +END)) DOCCLS(*ANY) OWNER(*CURRENT)

This command deletes all documents belonging to the current user.

Example 5: Deleting User's Documents Filed Today
DLTDLO DLO(*SEARCH)

This command deletes all documents belonging to the specified user that were filed on that day.

Example 6: Deleting Documents and Folders With Document Descriptors
DLTDLO DLO(*SEARCH) CRTDATE((AVAIL +BEGIN) (AVAIL +END)) DOCCLS(*ANY) OWNER(*ALL)

This command deletes all documents and folders with document descriptors from the system. If the document descriptors are not synchronized with the document and folder objects, then they are the same as DLO(*ALL). The Backup, Recovery, and Availability topic in the Information Center has more information about synchronizing document descriptors.

Error messages

*ESCAPE Messages

CPF2204
User profile &1 not found.

CPF8A16
Document library objects not deleted. &1 objects deleted.
CPF9005
   System resource required to complete this request not available.
CPF9006
   User not enrolled in system distribution directory.
CPF9012
   Start of document interchange session not successful for &1.
CPF9029
   Not allowed to specify owner profile &1.
CPF903A
   Document or folder activity not stopped, requested operation cannot be done.
CPF9031
   No authority to specify DLO(*ALL).
CPF9032
   Document interchange session not started.
CPF9046
   No documents found satisfying search specification in folder &1.
CPF9062
   Date must be specified when time not *AVAIL.
CPF9063
   Starting and ending CRTDATE specified in reverse order.
CPF9096
   Cannot use CMDCHRID(*DEVD), DOCCHRID(*DEVD) in batch job.
CPF9845
   Error occurred while opening file &1.
CPF9846
   Error while processing file &1 in library &2.
CPF9847
   Error occurred while closing file &1 in library &2.
Delete Document List (DLTDOCL)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Delete Document List (DLTDOCL) command allows you to delete a document list from the system. All document lists are stored in library QUSR SYS.

Restrictions:
- To delete a document list, you must have object existence (*OBJEXIST) authority for the document list, all object (*ALLOBJ) special authority, or security administrator (*SECADM) special authority.
- To delete a document list for another user, you must have *ALLOBJ or *SECADM special authority.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
</table>
| DOCL    | Document list        | Single values: *ALL
Other values (up to 300 repetitions): Name                               | Required, Positional 1       |
| OWNER   | Document list owner  | Element list                                                            | Optional, Positional 2       |
|         | Element 1: User profile | Name, *CURRENT, *ALL                                                 |                              |

Document list (DOCL)

Specifies the document lists that are to be deleted from the system.

*ALL  All document list objects for the specified owner are deleted.
name  Specify the name of the document list object that is to be deleted. A maximum of 300 document list objects can be specified.

Document list owner (OWNER)

Specifies the owner of the document lists that are to be deleted. This parameter is ignored if a list of document list names is specified on the Document list (DOCL) parameter.

*CURRENT  All document lists owned by you are to be deleted.
*ALL  All document lists for all owners are deleted.
name  Specify the name of the user who owns the document lists that are to be deleted. All document lists owned by this user are deleted.
Examples

Example 1: Deleting Current User’s Document Lists
DLTDOCL  DOCL(*ALL)  OWNER(*CURRENT)

This command deletes all document lists owned by the current user.

Example 2: Deleting All Users’ Document Lists
DLTDOCL  DOCL(*ALL)  OWNER(*ALL)

This command deletes all document lists owned by all users.

Example 3: Deleting Document Lists Owned By a Specified User
DLTDOCL  DOCL(*ALL)  OWNER(ANN)

This command deletes all document lists owned by ANN.

Example 4: Deleting a Specific Document List Owned By User
DLTDOCL  DOCL(AN8T475237)

This command deletes document list AN8T475237, owned by the user running this command. Displaying the user’s owned objects will show the system object name for a specific document list.

Error messages

*ESCAPE Messages

CPF9002
Not allowed to delete specified document lists.

CPF9009
System requires file &1 in &2 be journaled.

CPF9012
Start of document interchange session not successful for &1.

CPF9032
Document interchange session not started.

CPF9051
&1 document lists deleted; &2 document lists not deleted.

CPF9810
Library &1 not found.

CPF9845
Error occurred while opening file &1.

CPF9846
Error while processing file &1 in library &2.

CPF9847
Error occurred while closing file &1 in library &2.
Delete Distribution (DLTDST)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Delete Distribution (DLTDST) command allows you to delete your own distributions or to delete distributions for another user. This command also allows system operators to delete damaged distribution objects and dangling distribution objects (distribution objects that are not controlled by a distribution tracking object).

Restrictions:
1. If you are working on behalf of another user, you must have *ALLOBJ special authority or have been granted permission to work on behalf of the other user with the Grant User Permission (GRTUSRPMN) command.
2. The requester of the command must be enrolled in the system distribution directory.
3. Personal distribution cannot be requested if the requester is working on behalf of another user.
4. Damaged and dangling distribution objects can be deleted only by the user who signs on and has *ALLOBJ authority.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
</table>
| DSTID    | Distribution identifier | Single values: *ERROBJ  
Other values (up to 50 repetitions): Character value | Required, Positional 1 |
| OPTION   | Incoming or outgoing | *IN, *OUT, *ERR                                                        | Optional               |
| USRID    | User identifier      | Single values: *CURRENT  
Other values: Element list | Optional               |
|          | Element 1: User ID   | Character value                                                         |                       |
|          | Element 2: Address   | Character value                                                         |                       |
| DSTIDEXN | Distribution ID extension | Values (up to 50 repetitions): 0-99, *NONE  | Optional               |
| OBJ      | Object               | Single values: *NONE, *ALL  
Other values (up to 50 repetitions): Character value | Optional               |
| CMDCHRID | Command character identifier | Single values: *SYVAL, *DEVD  
Other values: Element list | Optional               |
|          | Element 1: Graphic character set | Integer                |                        |
|          | Element 2: Code page  | Integer                                                               |                        |
Distribution identifier (DSTID)

Specifies the unique identifier of the distribution. The identifier was assigned to the distribution by the system that originated it. Distribution identifiers can be found by using the Query Distribution (QRYDST) command. Identifiers are also returned from the Send Distribution (SNDDST) command.

You can enter multiple values for this parameter.

This is a required parameter.

*ERROBJ

The distributions to be deleted are damaged or dangling distribution objects.

Note: To delete a damaged or dangling distribution, a value must be specified also on the OBJ parameter.

distribution-ID

Specify the distribution identifiers. You can specify:

• up to a maximum of 50 identifiers
• sender’s address (insert blanks on right to total 8 characters)
• sender’s user ID (insert blanks on right to total 8 characters)
• zoned sequence number (4 digits with leading zeros)

Notes:

– Apostrophes are needed if blanks or special characters are in the distribution identifier.
– The distribution specified cannot be damaged or dangling.

Incoming or outgoing (OPTION)

Specifies whether the distribution named on the Distribution identifier prompt (DSTID parameter) is an incoming distribution, an outgoing distribution, or an error distribution.

*IN An incoming distribution sent to the user is deleted.

*OUT An outgoing distribution from the user is deleted. An outgoing distribution is the status being saved by the system for a distribution sent to one or more users with confirmation of delivery requested.

*ERR An error status distribution is deleted. This is the status returned because an error occurred during routing of the distribution.

User identifier (USRID)

Specifies which user ID and user ID address should be associated with the request.

*CURRENT

You are performing the request for yourself.

user-ID

Specify another user’s user ID or your user ID. You must have been given permission to work on behalf of another user or have *ALLOBJ authority.

user-ID-address

Specify another user’s address or your address. You must have been given permission to work on behalf of another user or have *ALLOBJ authority.
**Distribution ID extension (DSTIDEXN)**

Specifies the extension of the distribution identifier (if any) specified on the Distribution identifier prompt (DSTID parameter). This extension uniquely identifies duplicate distributions. This extension is a 2-digit extension that ranging from 00 through 99.

*NONE*

There is no duplicate distribution. *NONE* is equivalent to an extension of 01 for incoming distributions. For outgoing and undelivered status distributions, this is equivalent to 00.

**distribution-id-extension**

Specify the extension associated with the distribution. This is used to uniquely identify duplicate distributions.

---

**Object (OBJ)**

Specifies the damaged or dangling distribution objects to be deleted.

*NONE*

No damaged or dangling distribution objects are deleted.

*ALL*

All damaged and dangling distribution objects are deleted.

**object-name**

Specify the name of the damaged or dangling distribution object to be deleted.

The object named can be a local or remote object.

- Local object names consist of the sender’s address (padded on the right with blanks to a maximum of 8 characters), the sender’s user ID (padded on the right with blanks to a maximum of 8 characters), and a 4-digit zoned sequence number with leading zeros. For example:

  `’NEWYORK SMITH 0204’` or
  `UADDRESSUSERID0099`

  The apostrophes are needed if there are blanks or special characters in the distribution identifier. The distribution identifier is specified this way because blank characters are valid in a user ID or address.

- Remote object names consist of the remote system name and the system date and time. For example:

  `’QOSRDIST091112509152355’`

  where QOSRDIST is the remote system name, 0911125 is the system date (November 25, 1991), and 09152355 is the system time (9:15:23:55).

The distribution identifier for damaged or dangling distributions can be determined when the Save Document Library Object (SAVDLO) command is run and damaged or dangling distribution objects are identified.

A maximum of 50 objects can be named to be deleted.
Command character identifier (CMDCHRID)

Specifies the character identifier (graphic character set and code page) for the data being entered as command parameter values. The character identifier is related to the display device used to enter the command.

If the values specified on the Distribution identifier prompt (DSTID parameter) and User identifier prompt (USRID parameter) are being read from an output file created by the Query Distribution (QRYDST) command, specify ‘930 500’ on this parameter.

*SYSVAL
The system determines the graphic character set and code page values for the command parameters from the QCHRID system value.

*DEVD
The system determines the graphic character set and code page values from the display device description where this command was entered. This option is valid only when entered from an interactive job. If this option is specified in a batch job, an error occurs.

Element 1: Graphic character set

1-32767
Specify the graphic character set to use.

Element 2: Code page

1-32767
Specify the code page to use.

Examples

Example 1: Deleting Incoming Distribution Sent to User

DLTDST DSTID(‘RCH38P BAKER 0019’) USRID(*CURRENT) DSTIDEXN(02)

This command deletes an incoming distribution sent to the user.

Example 2: Deleting Status of Outgoing Distribution

DLTDST DSTID(‘SYSTEM20 BRUCE 1361’) USRID(*CURRENT) DSTIDEXN(00) OPTION(*OUT) CMDCHRID(101 37)

This command deletes the status associated with an outgoing distribution. The DSTID parameter value is encoded with a character identifier where the code page is 37 and the character set is 101. The system performs translation on the DSTID parameter value before trying to locate the distribution on the distribution recipient index.

Example 3: Deleting Incoming Distribution to Signed-On User

DLTDST DSTID(‘CHICAGO JONES 0013’)

This command deletes an incoming distribution sent to a user who is signed on. The distribution extension defaults to ‘01’ for incoming distributions. If duplicate distributions were sent, this command would only delete the first one.

Example 4: Deleting All Damaged Distributions
This command deletes all damaged and dangling distributions on the system.

**Error messages**

*ESCAPE Messages*

CPF89BD
The object must be specified with DSTID(*ERROBJ).

CPF900A
&2 distribution deleted, could not delete &1 distribution.

CPF900B
User ID and address &1 &2 not in System Distribution Directory.

CPF900C
Sign on and verify of user failed.

CPF9016
Request to cancel distribution &2-&3-&4 unsuccessful for &1.

CPF905C
Error occurred trying to find a translation table.

CPF906C
Distribution not deleted.

CPF9096
Cannot use CMDCHRID(*DEVD), DOCCHRID(*DEVD) in batch job.
IBM Systems – iSeries: i5/OS Commands Starting with CRTMSGF (Create Message File)
Delete Distribution List (DLTDSTL)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Delete Distribution List (DLTDSTL) command deletes an existing distribution list from the system distribution directory.

Restriction: You must have security administrator authority to delete a distribution list owned by another user. No special authority is needed to delete a distribution list that you own.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>LSTID</td>
<td>List identifier</td>
<td>Values (up to 300 repetitions): Element list</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Element 1: List ID</td>
<td>Character value</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: List ID qualifier</td>
<td>Character value</td>
<td></td>
</tr>
<tr>
<td>CMDCHRID</td>
<td>Command character identifier</td>
<td>Single values: *SYSVAL, *DEVD</td>
<td>Optional</td>
</tr>
<tr>
<td></td>
<td>Element 1: Graphic character set</td>
<td>Element list</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Element 2: Code page</td>
<td>Integer</td>
<td></td>
</tr>
</tbody>
</table>

List identifier (LSTID)

Specifies the two-part list identifier of the distribution list being deleted. A maximum of 300 list IDs (both parts) can be specified. You can specify a maximum of 8 characters for each part of each list ID.

The possible list identifier value is:

list-ID

Specify the list identifier (ID) of the distribution list.

The possible list qualifier value is:

list-ID-qualifier

Specify the list ID qualifier of the distribution list.

Note: The distribution list identifier has two parts, the ID and the qualifier, separated by at least one space. If lowercase characters are specified, the system changes them to uppercase.

The naming rules for the two-part list ID are identical to the rules for the user ID and address. A complete description of these rules is in the SNA Distribution Services book, SC41-5410.

You can enter multiple values for this parameter.

This is a required parameter.
Command character identifier (CMDCHRID)

Specifies the character identifier (graphic character set and code page) for the data being entered as command parameter values.

If the value for the Distribution identifier (DSTID) parameter and User identifier (USRID) parameter are being read from an output file created by the Query Distribution (QRYDST) command, use ‘930 500’ for the value of this parameter.

Note:
- Only the user ID and address, system name and group, department, and the X.400 O/R parameters are translated to the graphic character set identifier (GCID) specified on this parameter. All other parameter values that you specify are stored exactly as they are entered; the GCID value is stored with them.
- If this command is run interactively, the default GCID value is taken from the display device description. If it is run in batch, the default GCID value is taken from the QCHRID system value. You can override these values by specifying a specific character set and code page on this parameter.

*SYSVAL
  The system determines the graphic character set and code page values for the command parameters from the QCHRID system value.

*DEVD
  The system determines the graphic character set and code page values from the display device description where this command was entered. This option is valid only when entered from an interactive job. If this option is specified in a batch job, an error occurs.

Element 1: Graphic character set

1-32767
  Specify the graphic character set to use.

Element 2: Code page

1-32767
  Specify the code page to use.

Examples

DLTDSTL LSTID((DEPT48K DLIST) (ISMGRS DSTLIST))

This command deletes the two distribution lists, DEPT48K DLIST and ISMGRS DSTLIST, if they exist.

Error messages

*ESCAPE Messages

CPF9024
  System cannot get correct record to finish operation.
CPF905C
   Error occurred trying to find a translation table.

CPF9094
   &1 distribution lists deleted. &2 lists not deleted.

CPF9096
   Cannot use CMDCHRID(*DEVD), DOCCHRID(*DEVD) in batch job.

CPF9838
   User profile storage limit exceeded.

CPF9845
   Error occurred while opening file &1.

CPF9846
   Error while processing file &1 in library &2.
Delete Data Area (DLTDTAARA)

Where allowed to run: All environments (*ALL)
Threadsafe: Yes

The Delete Data Area (DLTDTAARA) command deletes the specified data areas from a library.

Restrictions:
1. To use this command, you must have object existence (*OBJEXIST) authority for the data area, and execute (*EXECUTE) authority for the library.
2. Local data areas, group data areas, and program initialization parameter data areas cannot be deleted.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>DTAARA</td>
<td>Data area</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Data area</td>
<td>Generic name, name</td>
<td></td>
</tr>
</tbody>
</table>

Data area (DTAARA)

Specifies the name and library of the data areas being deleted. A specific data area or a generic data area can be specified; either type can be optionally qualified by a library name.

This is a required parameter.

Qualifier 1: Data area

generic-name

Specify the generic name of the data area being deleted. A generic name is a character string that contains one or more characters followed by an asterisk (*). If a generic name is specified, then all data areas that have names with the same prefix as the generic data area name are deleted.

name

Specify the name of the data area.

Qualifier 2: Library

*LIBL

All libraries in the thread’s library list are searched until a match is found. If a specific object name is specified (instead of a generic name), only the first object found to have that name is deleted.

*CURLIB

The current library for the thread is used to locate the object. If no library is specified as the current library for the thread, the QGPL library is used.
*USRLIBL
Only the libraries listed in the user portion of the library list are searched. If a specific object name is specified (instead of a generic name), only the first object found with that name is deleted.

*ALL
All libraries in auxiliary storage pools (ASPs) that are currently part of the thread’s library name space will be searched. This includes the system ASP (ASP 1), all defined basic user ASPs (ASPs 2-32), and, if the thread has an ASP group, the primary and secondary ASPs in the thread’s ASP group. Only your own QTEMP library is searched. All objects matching the specified name and object type in all libraries in the thread’s name space are deleted.

*ALLUSR
All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:

1. **CGULIB**
2. **DSULIB**
3. **SEULIB**
4. **COBLIB**
5. **RPGLIB**
6. **DFULIB**
7. **DALLIB**

Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered user libraries and are also searched:

- QDSNX
- QRCLxxxxx
- QUSRIJS
- QUSRvRxMx
- QGPL
- QSRVAGT
- QUSRINFSKR
- QGPL38
- QSYS2
- QUSRPOSGS
- QMPGDATA
- QUSER3B
- QUSRPOSSA
- QMQMDATA
- QUSRINFSKR
- QMQMPROC
- QUSRDIRCL
- QUSRPOSSA
- QMQMPROC
- QUSRDIRDB
- QUSRPOSSA
- QUSRvRxMx

1. ‘xxxxx’ is the number of a primary auxiliary storage pool (ASP).
2. A different library name, in the format QUSRvRxMx, can be created by the user for each previous release supported by IBM to contain any user commands to be compiled in a CL program for the previous release. For the QUSRvRxMx user library, VxRxMx is the version, release, and modification level of a previous release that IBM continues to support.

**name** Specify the library where the data area is located.

---

**Examples**

**DLTDTAARA**

DTAARA(MYLIB/MYDATA)

This command deletes the data area named MYDATA from the library MYLIB if the user has the proper authority for the data area and the library.

---

**Error messages**

**ESCAPE Messages**

**CPF2105**

Object &1 in &2 type *&3 not found.

**CPF2110**

Library &1 not found.
CPF2113
Cannot allocate library &1.

CPF2114
Cannot allocate object &1 in &2 type *&3.

CPF2117
&4 objects type *&3 deleted. &5 objects not deleted.

CPF2176
Library &1 damaged.

CPF2182
Not authorized to library &1.

CPF2189
Not authorized to object &1 in &2 type *&3.
Delete Data Dictionary (DLTDTADCT)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Delete Data Dictionary (DLTDTADCT) command deletes a data dictionary. All program described files linked to definitions in the dictionary must be unlinked before the dictionary is deleted.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>DTADCT</td>
<td>Data dictionary</td>
<td>Name</td>
<td>Required, Positional 1</td>
</tr>
</tbody>
</table>

Data dictionary (DTADCT)
Specifies the name of the data dictionary to be deleted.
This is a required parameter.

Examples

DLTDTADCT   DTADCT(DEPT547)

This command deletes the DEPT547 data dictionary.

Error messages

*ESCAPE Messages

CPF2FE2
Dictionary &1 currently in use.

CPF2FE3
System cross reference file is in error.

CPF2FE4
System cross reference file not available.

CPF2F0B
Data dictionary &1 cannot be deleted.

CPF2F0C
Deleting a SQL data dictionary &1 not allowed.
CPF2F0D
  Data dictionary &1 partially deleted.

CPF2F0F
  Data dictionary &1 not deleted.

CPF2F05
  Not authorized to delete dictionary &1.

CPF2F08
  Dictionary &1 not found.

CPF9820
  Not authorized to use library &1.
Delete Data Queue (DLTDTAQ)

Where allowed to run: All environments (*ALL)
Threadsafe: Yes

The Delete Data Queue (DLTDTAQ) command deletes the specified data queue(s) from the system.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>DTAQ</td>
<td>Data queue</td>
<td>Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Data queue</td>
<td>Generic name, name</td>
<td></td>
</tr>
</tbody>
</table>

Data queue (DTAQ)

Specifies one or more data queues to be deleted.

This is a required parameter.

Qualifier 1: Data queue

name Specify the name of the data queue to be deleted.

generic*name Specify the generic name of the data queue to be deleted. A generic name is a character string that contains one or more characters followed by an asterisk (*). If a generic name is specified, all data queues that have names with the same prefix as the generic data queue name are deleted.

Qualifier 2: Library

*LIBL All libraries in the library list for the current thread are searched until the first match is found.

*CURLIB The current library for the job is searched. If no current library exists in the library list, QGPL is used to locate the data queue to be deleted.

*USRLIBL If a current library entry exists in the library list for the current thread, the current library and the libraries in the user portion of the library list are searched. If there is no current library entry, only the libraries in the user portion of the library list are searched.

*ALL All libraries in the system, including QSYS, are searched.

*ALLUSR All user libraries are searched. All libraries with names that do not begin with the letter Q are searched except for the following:
Although the following Qxxx libraries are provided by IBM, they typically contain user data that changes frequently. Therefore, these libraries are considered user libraries and are also searched:

QDSNX  QRCXXX  QUSRIJS  QUSRVxRxMx
QGPL   QSRVAGT  QUSRINFSKR
QGPL38 QSYS2    QUSRNOTES
QMGTC  QSYS2xxxxx  QUSROND
QMGTC2 QS36F    QUSRPOSGS
QMPDATA QUSER38  QUSRPOSSA
QMMDATA QUSRADSM  QUSRPMYSVR
QMMPROC QUSRBRM  QUSRPOSDS
QPFDATA QUSRDIRCL  QUSRPOSSA
QRCL   QUSRDIRDB  QUSRPOSGS

1. ‘xxxxx’ is the number of a primary auxiliary storage pool (ASP).
2. A different library name, in the format QUSRVxRxMx, can be created by the user for each previous release supported by IBM to contain any user commands to be compiled in a CL program for the previous release. For the QUSRVxRxMx user library, VxRxMx is the version, release, and modification level of a previous release that IBM continues to support.

name Specify the library to be searched.

Examples

DLTDTAQ DTAQ(DEPTADTAQ)

This command deletes the data queue named DEPTADTAQ from the system.

Error messages

*ESCAPE Messages

CPF2105
Object &1 in &2 type *&3 not found.

CPF2110
Library &1 not found.

CPF2113
Cannot allocate library &1.

CPF2117
&4 objects type *&3 deleted. &5 objects not deleted.

CPF2182
Not authorized to library &1.

CPF2189
Not authorized to object &1 in &2 type *&3.

CPF9503
Cannot lock data queue &1 in &2.
Delete Edit Description (DLTEDTD)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Delete Edit Description (DLTEDTD) command deletes a specified user-defined edit description.

Note: Any data description specifications (DDS), or high-level language programs that have already been created, are not affected.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>ETDTD</td>
<td>Edit description</td>
<td>Single values: 5, 6, 7, 8, 9 Other values: Qualified object name</td>
<td>Required, Positional 1</td>
</tr>
<tr>
<td></td>
<td>Qualifier 1: Edit description</td>
<td>Name</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Qualifier 2: Library</td>
<td>Name, QSYS</td>
<td></td>
</tr>
</tbody>
</table>

Edit description (EDTD)

Specifies a single-digit code (5, 6, 7, 8, or 9) or the name of the user-defined edit description being deleted. If a single-digit code is entered, the IBM-supplied edit description name corresponding to the digit is assumed. For example, if the digit 5 is entered, the IBM-supplied edit description name QEDIT5 is assumed.

This is a required parameter.

The possible values are:

- **single-digit-code**
  - Specify the digit code (5, 6, 7, 8, or 9) for the edit description to be deleted.

- **edit-description-name**
  - Specify the name of the edit description to be deleted.

The possible library values are:

- **QSYS** Only the QSYS library is searched for the specified edit description.

- **library-name**
  - Specify the library to be searched for the specified edit description.
Examples

DLTEDTO  EDTD(5)

This command deletes the user-defined edit description 5 from the system.

Error messages

*ESCAPE Messages

CPF2105
Object &1 in &2 type *&3 not found.

CPF2110
Library &1 not found.

CPF2114
Cannot allocate object &1 in &2 type *&3.

CPF2182
Not authorized to library &1.

CPF2189
Not authorized to object &1 in &2 type *&3.

CPF2625
Not able to allocate object &1.
Delete Expired Spooled files (DLTEXPSPLF)

Where allowed to run: All environments (*ALL)
Threadsafe: No

The Delete Expired Spooled Files (DLTEXPSPLF) command removes expired spooled files on the system or the auxiliary storage pool (ASP) group specified by the user. A spooled file expires at 23:59:59, system local time on the date specified in the Expiration date for file (EXPDATE) spooled file attribute. To automate the removal of expired spooled files, you can create a job schedule entry that runs the DLTEXPSPLF command once a day. See the Examples section for an example using the Add Job Schedule Entry (ADDJOBSCDE) command.

Restriction:
- You need spool control (*SPLCTL) special authority to use this command. If *ALL or a specific ASP group (ASPGRP) is specified, the user must be authorized to the ASP group and the status of the ASP group must be AVAILABLE.

Parameters

<table>
<thead>
<tr>
<th>Keyword</th>
<th>Description</th>
<th>Choices</th>
<th>Notes</th>
</tr>
</thead>
<tbody>
<tr>
<td>ASPGRP</td>
<td>ASP group</td>
<td>Name, *ALL, *SYSBAS</td>
<td>Optional, Positional 1</td>
</tr>
</tbody>
</table>

ASP group (ASPGRP)

Specifies the auxiliary storage pool (ASP) group from which expired spooled files are to be removed.

*ALL          Expired spooled files which are found in the system ASP (ASP 1), all defined basic user ASPs (ASPs 2-32) and all primary and secondary ASPs are deleted.
*SYSBAS        Expired spooled files which are found in the system ASP (ASP 1) and all defined basic user ASPs (ASPs 2-32) are deleted.
name           Specify the name of a primary ASP group. Expired spooled files which are found in the primary and secondary ASPs of the specified ASP group are deleted.

Examples

Example 1: Deleting All Expired Spooled Files
DLTEXPSPLF  ASPGRP(*ALL)

This command deletes all expired spooled files on the system.
Example 2: Deleting All Expired Spooled Files in the System ASP (ASP 1) and All Defined Basic User ASPs (ASPs 2-32)

DLTEXTXPSPLF  ASPGRP(*SYSBAS)

This command deletes all expired spooled files in the system ASP (ASP 1) and all defined basic user ASPs (ASPs 2-32).

Example 3: Creating a Job Schedule Entry to Remove Expired Spooled Files

ADDJOBSCDE  JOB(DLTEXTXPSPLF)  CMD(DLTEXTXPSPLF  ASPGRP(*ALL))
FRQ(*WEEKLY)  SCDDATE(*NONE)  SCDDAY(*ALL)
SCDTIME(010000)  JOBO(QSYS/QSYSSNOMAX)
TEXT('DELETE EXPIRED SPOOLED FILES SCHEDULE ENTRY')

This command creates a job schedule entry which causes the DLTEXTXPSPLF command to run daily at the time specified in the SCDTIME parameter in the job queue specified in the JOBO parameter. Using the FRQ(*WEEKLY) and SCDDAY(*ALL) parameters will schedule the job to run every day of the week.

Error messages

*ESCAPE Messages

CPF3330
   Necessary resource not available.

CPF334F
   Not authorized to command Delete Expired Spooled files (DLTEXTXPSPLF).

CPF9825
   Not authorized to device &1.

CPFB8E9
   ASP group &1 not set for thread &2.
Appendix. Notices

This information was developed for products and services offered in the U.S.A.

IBM may not offer the products, services, or features discussed in this document in other countries. Consult your local IBM representative for information on the products and services currently available in your area. Any reference to an IBM product, program, or service is not intended to state or imply that only that IBM product, program, or service may be used. Any functionally equivalent product, program, or service that does not infringe any IBM intellectual property right may be used instead. However, it is the user's responsibility to evaluate and verify the operation of any non-IBM product, program, or service.

IBM may have patents or pending patent applications covering subject matter described in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to:

IBM Director of Licensing
IBM Corporation
500 Columbus Avenue
Thornwood, NY8809
U.S.A.

For license inquiries regarding double-byte (DBCS) information, contact the IBM Intellectual Property Department in your country or send inquiries, in writing, to:

IBM World Trade Asia Corporation
Licensing
2-31 Roppongi 3-chome, Minato-ku
Tokyo 106, Japan

The following paragraph does not apply to the United Kingdom or any other country where such provisions are inconsistent with local law: INTERNATIONAL BUSINESS MACHINES CORPORATION PROVIDES THIS PUBLICATION “AS IS” WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF NON-INFRINGEMENT, MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. Some states do not allow disclaimer of express or implied warranties in certain transactions, therefore, this statement may not apply to you.

This information could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or the program(s) described in this publication at any time without notice.

Any references in this information to non-IBM Web sites are provided for convenience only and do not in any manner serve as an endorsement of those Web sites. The materials at those Web sites are not part of the materials for this IBM product and use of those Web sites is at your own risk.

IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Licensees of this program who wish to have information about it for the purpose of enabling: (i) the exchange of information between independently created programs and other programs (including this one) and (ii) the mutual use of the information which has been exchanged, should contact:

IBM Corporation

© Copyright IBM Corp. 1998, 2006
Such information may be available, subject to appropriate terms and conditions, including in some cases, payment of a fee.

The licensed program described in this information and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, or any equivalent agreement between us.

Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products.

This information contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

COPYRIGHT LICENSE:

This information contains sample application programs in source language, which illustrate programming techniques on various operating platforms. You may copy, modify, and distribute these sample programs in any form without payment to IBM, for the purposes of developing, using, marketing or distributing application programs conforming to the application programming interface for the operating platform for which the sample programs are written. These examples have not been thoroughly tested under all conditions. IBM, therefore, cannot guarantee or imply reliability, serviceability, or function of these programs. You may copy, modify, and distribute these sample programs in any form without payment to IBM for the purposes of developing, using, marketing, or distributing application programs conforming to IBM’s application programming interfaces.

Each copy or any portion of these sample programs or any derivative work, must include a copyright notice as follows:

(C) IBM 2006. Portions of this code are derived from IBM Corp. Sample Programs. (C) Copyright IBM Corp. 1998, 2006. All rights reserved.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

**Trademarks**

The following terms are trademarks of International Business Machines Corporation in the United States, other countries, or both:

- Advanced Function Printing
- AFP
- AS/400
- CICS
- COBOL/400
- C/400
- DataPropagator
- DB2
IBM
Infoprint
InfoWindow
iSeries
LPDA
OfficeVision
i5/OS
Print Services Facility
RPG/400
SystemView
System/36
TCS
WebSphere

Microsoft, Windows, Windows NT, and the Windows logo are trademarks of Microsoft Corporation in the United States, other countries, or both.

Java and all Java-based trademarks are trademarks of Sun Microsystems, Inc. in the United States, other countries, or both.

UNIX is a registered trademark of The Open Group in the United States and other countries.

Other company, product, and service names may be trademarks or service marks of others.

Terms and conditions
Permissions for the use of these publications is granted subject to the following terms and conditions.

Personal Use: You may reproduce these publications for your personal, noncommercial use provided that all proprietary notices are preserved. You may not distribute, display or make derivative works of these publications, or any portion thereof, without the express consent of IBM.

Commercial Use: You may reproduce, distribute and display these publications solely within your enterprise provided that all proprietary notices are preserved. You may not make derivative works of these publications, or reproduce, distribute or display these publications or any portion thereof outside your enterprise, without the express consent of IBM.

Except as expressly granted in this permission, no other permissions, licenses or rights are granted, either express or implied, to the publications or any information, data, software or other intellectual property contained therein.

IBM reserves the right to withdraw the permissions granted herein whenever, in its discretion, the use of the publications is detrimental to its interest or, as determined by IBM, the above instructions are not being properly followed.

You may not download, export or re-export this information except in full compliance with all applicable laws and regulations, including all United States export laws and regulations.

IBM MAKES NO GUARANTEE ABOUT THE CONTENT OF THESE PUBLICATIONS. THE PUBLICATIONS ARE PROVIDED "AS-IS" AND WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE.