System i

Programming
i5/OS commands
Starting with SAVS36F (Save S/36 File)

Version 6 Release 1

System i

Programming
i5/OS commands
Starting with SAVS36F (Save S/36 File)

Version 6 Release 1

Note

Before using this information and the product it supports, be sure to read the information in "Notices," on page 629.

This edition applies to version 6, release 1, modification 0 of IBM i5/OS (product number 5761-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.

Contents

##

##

##

##

##

##

##

##

##

Start S/36 Session (STRS36) 625

Appendix. Notices 629

Save S/36 File (SAVS36F)

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

Parameters
Examples
Error messages

The Save System/36 File (SAVS36F) command can create:

- a copy of a single database physical file or logical file to diskette, tape magnetic media or another database physical file in the same format as if a System/36 has saved the file.
- a copy of multiple database physical files or logical files to diskette or tape magnetic media in the same format as if a System/36 had saved the files as a save all set. The files can be all files in a library, all files that are not part of a file group, all files in a specific file group, or all files that begin with a specified set of characters.

A **Save All Set** is a group (set) of files that share the same group (set) name and that are saved (copied) to diskette or tape with one operation. The set of files can be restored (copied back from diskette or tape) with a single operation by referring to the set name (see the Restore System/36 Files (RSTS36F) command).

File groups are defined by file names that contain a period. The characters preceding the period identify the file group, and the characters following the period identify the file within the group. As with file names within the System/36 environment, the maximum number of characters is eight, including the period. Files with names that do not contain a period are not part of a file group. The following examples show the names of files within a file group.

```
PAYROL.A
PAYROL.B    Files in File Group PAYROL
PAYROL.C
A.ACCTS
A.INV
A.PROL      Files in File Group A
A.B.GO
A.B.INV
A.B.GO
A.B.INV     Files in File Group A.B
```

The saved files can be restored to the following systems:

- System/36 (RESTORE procedure or $COPY utility)
- i5/OS (Restore System/36 File (RSTS36F) command)

The SAVS36F command is intended for exchanging files with a System/36. For creating a backup version of a file, the i5/OS save commands (for example, Save Object (SAVOBJ) or Save Changed Object (SAVCHGOBJ)) should be used.

Restrictions:

1. The following authorities are required (normally only applies when running on a system using resource security):
   - *USE authority for this command.
   - *USE authority for the file or group of files specified in the FROMFILE parameter.
   - *USE authority for the library specified in the FROMLIB parameter.

- *CHANGE authority to the file specified on the PHYFILE parameter if saving to an existing physical file.
- *USE authority for the library specified in the PHYFILE parameter if saving to a physical file.
- *CHANGE authority for the library specified in the PHYFILE parameter if saving to a physical file and the file does not exist.
- *USE authority for the diskette device description object, *USE authority for device file QSYSDKT, in library QSYS if saving to diskette.
- *USE authority for the tape device description object, *USE authority for device file QSYSTAP, in library QSYS if saving to tape.
- *USE authority for the based-on physical file if saving a logical file.

2. All diskettes that are used for the save operation should be initialized using the INZDKT CL command or the equivalent System/36 environment function (INIT operator control language (OCL) procedure or $INIT SSP utility). For a two-sided diskette, use a sector size of 256 or 1024. For a one-sided diskette, use a sector size of 128 or 512. If tape is used, each tape volume used should have been initialized with standard labels using the INZTAP CL command or the equivalent System/36 environment function (TAPEINIT OCL procedure or $TINIT SSP utility). Use a density of 1600 bits per inch when initializing the tape.

   Note: If the tape or diskette has not been initialized as stated above, the System/36 will not be able to process the media.
3. Object-level and record-level functions, other than read operations, should not be attempted for a file being saved by SAVS36F. Concurrent activity against the file (for example, moving the file or adding or removing records) can cause:
   - For a save operation of a single file (FROMFILE(file-name)), the save operation will end with escape message CPF9826 because the file cannot be allocated.
   - For a save operation of multiple files (FROMFILE(*ALL or generic*-file-name)), the save function sends an inquiry message CPA2C6A because the file cannot be allocated. The message allows an ignore, retry and cancel response. The ignore response bypasses this file and attempts to save the next file selected.
4. When saving a single file to diskette, the diskette cannot already contain an active file with the same label and creation date as the new file to be created.
5. When saving multiple files to diskette, the diskette used for the save cannot contain any active files.
6. Not all physical and logical files can be saved with the SAVS36F command.
   - Only logical files created under the System/36 environment (for example, through the BLDINDEX OCL procedure) or through a DDM request from a System/36 system can be saved. These files are saved as System/36 alternative index files.
   - All physical files created under the System/36 environment (for example, through the BLDFILE OCL procedure) or through a DDM request from a System/36 system are saved using information stored within the i5/OS file description. These files are saved as System/36 sequential, direct, or indexed physical files.
   - Any physical files created by i5/OS commands or utilities can be saved as long as the record length is not greater than 4096. These files are saved as System/36 sequential files.
7. To generate a save format which can be processed by the System/36 RESTORE procedure, the following information is not saved:
   - If saving a logical file, only the description of the file is saved. The index (or access path) is not saved.
   - If saving an indexed (keyed) physical file, the data is saved but the index is not. The index will be rebuilt after the file is restored.
8. The following restrictions apply to naming standards:
   - When saving a single file, the specified name (FROMFILE parameter) must meet naming standards. If not, message CPF0001 is sent when the SAVS36F command is processed.

- If a file name is found during a save operation of multiple files (FROMFILE(*ALL or generic*-file-name)) that does not meet the System/36 naming standards, diagnostic message CPF2C0E is sent and the file is not saved.

9. Multiple files (FROMFILE(*ALL) or FROMFILE(generic*-name)) cannot be saved to a physical file.

Top

Parameters

Keyword	Description	Choices	Notes
FROMFILE	From file	*Generic name, name*, *ALL	Required, Positional 1
FROMLIB	From library	*Name*	Required, Positional 2
DEV	Device	Single values: *PHYFILE Other values (up to 4 repetitions): *Name*	Required, Positional 3
GROUP	File group qualifier	*Character value*, ***ALL**, *NONE	Optional
SET	Set identifier	*Character value*, **#SAVE**	Optional
TOLABEL	File label	*Character value*	Optional
SEQNBR	Sequence number	1-9999, ***END**	Optional
VOL	Volume identifier	Values (up to 50 repetitions): *Character value*, ***MOUNTED**	Optional
RETAIN	Retention period	0-999, **1**	Optional
ENDOPT	End of tape option	***REWIND**, *LEAVE, *UNLOAD	Optional
DTACPR	Data compression	*YES, ***NO**	Optional
PHYFILE	Physical file	*Qualified object name*	Optional
	Qualifier 1: Physical file	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
MBROPT	Data base member option	***NOREPLACE**, *REPLACE	Optional
CRTDATE	Creation date	*Date*, ***LAST**, *ALL	Optional

Top

From file (FROMFILE)

Specifies the name of the file being saved. The file specified must exist in the library specified in the **From library** prompt (FROMLIB parameter).

This is a required parameter.

file-name
Specify the name of a single file to save.

***ALL** All files in the specified library are saved. To further describe which files are saved, use the **File group qualifier** prompt (GROUP parameter).

generic*-file-name
Specify a generic name of a group of files in the specified library to save. All files with the same prefix as the generic name are saved. A generic name is specified as a character string of one or more characters, followed by an asterisk.

Top

From library (FROMLIB)

Specifies which library contains the database files to be saved.

This is a required parameter.

Top

Device (DEV)

Specifies the name of a diskette unit or the names of one or more tape devices. A maximum of four tape device names can be specified. If more than one tape device is used, enter the names of the devices in the order in which they are used. Each device name must be already known on the system by a device description.

This is a required parameter.

***PHYFILE**
A database physical file receives the copied file. The qualified name of the physical file must be specified on the **Physical file** prompt (PHYFILE parameter). *PHYFILE is not valid if *ALL or a generic*-file-name is specified for the **From file** prompt (FROMFILE parameter).

device-name
Specifies the name of the diskette unit or the names of one or more tape devices used for the save operation.

Top

File group qualifier (GROUP)

Specifies which file groups are to be saved. This parameter is valid only if *ALL is specified for the **From file** prompt (FROMFILE parameter).

***ALL** All files are saved.

***NONE**
No files that belong to a file group are saved. Only files that do not belong to file group are saved.

group-name
Specify the name of a file group. All files that belong to that file group are saved. Files that do not belong to the specified file group are not saved. A group name can be up to 7 characters in length. The first character in the name must be an alphabetic character (A through Z, #, @, or $). The remaining characters can be any combination of characters (numeric, alphabetic, and special) except commas (,), apostrophes (′), quotation marks (″), question mark (?), asterisk (*), or blanks. The period, which indicates a file group name, must not be included as part of the group name.

Top

Set identifier (SET)

Specifies the set identifier used to identify the entire set of files to be saved. This parameter is valid only if *ALL or a generic name is specified for the **From file** prompt (FROMFILE parameter).

#SAVE
The default set identifier is used.

set-identifier
Specify the set identifier used to identify the entire set of files to be saved. The set identifier can be up to 8 characters in length. The first character in the name must be an alphabetic character (A through Z, #, @, or $). The remaining characters can be any combination of characters (numeric, alphabetic, and special) except commas (,), apostrophes (′), quotation marks (″), question mark (?), asterisk (*), or blanks.

Top

File label (TOLABEL)

Specifies the label value (eight characters maximum) given to the new diskette or tape file created by the save operation of a single file. If no value is specified, the value of the **From file** prompt (FROMFILE parameter) is used as the diskette or tape label.

This parameter is not allowed if *ALL or a generic name is specified for **From file** prompt (FROMFILE parameter).

Top

Sequence number (SEQNBR)

Specifies, only when tape is used, which sequence number is used for the save operation.

***END** The specified file or files are saved starting after the last sequence number on the first tape (that is, this file follows all other files currently on the tape volume). If the first tape is full, an error message is issued and the operation ends. If the sequence number to be assigned to the specified file is greater than 9999, an error message is issued and the operation ends.

If multiple files are saved, the next file is saved to a file after the first file that was saved, and so on. If the sequence number to be assigned ever exceeds 9999, an error message is issued and the operation ends.

file-sequence-number
Specify the sequence number of the tape file that is used for the save operation. Valid values range from 1 through 9999.

If this sequence number already exists on the tape volume, the tape label at that sequence number must match the TOLABEL parameter. The existing file at that sequence number is overwritten, and all subsequent files on the volume are not accessible after the save.

If a new tape file is added to the tape, the sequence number must be one higher than the sequence number of the last tape file on that volume. No gaps are allowed in the series of sequence numbers.

If multiple files are being saved, this sequence number is used for the first file. All remaining files are saved as if *END was specified on the parameter SEQNBR. If the sequence number to be assigned ever exceeds 9999, an error message is issued and the operation ends.

Top

Volume identifier (VOL)

Specifies the volume identifiers of the tape volumes on which the object data is to be saved. The volumes must be placed in the device in the same order as the volume identifiers are specified for this parameter.

***MOUNTED**
The volume currently placed in the device is used.

volume-identifier
Specify the volume identifiers of the tapes or diskettes used for the save operation. A maximum of 50 volume identifiers can be specified.

Top

Retention period (RETAIN)

Specifies the retention period for the newly created tape or diskette file. The file is protected and cannot be written over until the day after the retention period ends.

1 A retention period of one day is used.

retention-period
Specify the number of days the tape or diskette file should be kept. If a retention period of 999 is specified, the tape or diskette file becomes a permanent file.

Top

End of tape option (ENDOPT)

Specifies, only when tape is used, what operation is automatically performed on the tape volume after the save operation ends. This parameter applies only to the last reel used.

***REWIND**
The tape is rewound, but not unloaded.

***LEAVE**
The tape is not rewound.

***UNLOAD**
The tape is automatically rewound and unloaded after the operation ends.

Top

Data compression (DTACPR)

Specifies, when a diskette is used, whether the data is compressed into System/36 compatible format before it is written to the diskette. If the save command is operating while other jobs on the system are active and data compression is used, the overall system performance may be affected. This parameter is not valid if *PHYFILE or a tape device is specified on the **Device** prompt (DEV parameter).

The possible values are:

***NO** The data is not compressed before being written to the diskette.

***YES** The data is compressed before being written to the diskette.

Top

Physical file (PHYFILE)

Specifies the name of the file that receives the copied file. If the specified file does not exist, it is created in the current library as a non-keyed, program-described file with a record length of 256. The copied records are put in the first member of the physical file. If the file has no members, a member is created using the system date.

The possible library values are:

***LIBL** The library list 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, QGPL is used.

library-name
Specify the library where the file is located.

Top

Data base member option (MBROPT)

Specifies whether the new records replace or are added to the existing records.

***NOREPLACE**
Specifies whether a file already exists by the name specified on the PHYFILE parameter in the specified library, an error message is sent and the data in that member is not replaced.

***REPLACE**
The PHYFILE member is cleared before copying the first record.

Top

Creation date (CRTDATE)

Specifies, for a *date-differentiated* file (maintained by the System/36 environment), which instance (member) of the file or files is saved. A date-differentiated file has the same name as another file but it has a different file creation date. On this system, date-differentiated files are multiple-member physical files.

***LAST**
The most recently created member for the specified file or files are saved.

***ALL** All members in the date-differentiated file are saved. If the file being saved is not date differentiated, only the last member created in the file is saved. *ALL is valid only when *ALL or a generic file name is specified for **From file** prompt (FROMFILE parameter).

file-creation-date
Specify the creation date of the date-differentiated file member to save. A file creation date is valid only when a single file is saved, or a file name is specified on the **From file** prompt (FROMFILE parameter).

Top

Examples

Example 1: Saving a Single File

```
SAVS36F   FROMFILE(PETE)  FROMLIB(QS36F)  DEV(I1)
```

This command saves the file named PETE located in library QS36F. Assuming that I1 is the name of a diskette device description, the file is saved on the diskette placed in the diskette drive. The diskette file label is PETE (same as the FROMFILE name). If PETE is a date-differentiated physical file, the most recently created instance (member) of PETE is saved. The diskette file has a retention period of one day (the retention period ends at midnight of the following day).

Example 2: Saving a Single File

```
SAVS36F   FROMFILE(MSTRPAY)  FROMLIB(PAYLIB)  DEV(T1 T2)
          TOLABEL('PAY.MSTR')  RETAIN(999)
```

This command saves the file named MSTRPAY located in library PAYLIB. Assuming that T1 and T2 are tape devices, the file is copied to the tapes on devices T1 and T2. The tape file label is PAY.MSTR and the tape file is a permanent file. The last tape used for the save is rewound at the end of the save operation.

Example 3: Saving Multiple Files

```
SAVS36F   FROMFILE(*ALL)  FROMLIB(QS36F)  DEV(T1 T2)
          GROUP(*ALL)  SET(ALLFL)  RETAIN(999)
```

This command saves all database physical and logical files in library QS36F (including all files that belong to a file group). If any of the files are date-differentiated files, only the last member created in each file is saved. Assuming that T1 and T2 are tape devices, the files are copied to the tape volumes that are placed in tape drives T1 and T2. The label of the tape files created are the same as the names of the files that are saved. The first tape file created is located after the last sequence number on the tape. The remaining files are located after that first file. The tape files created are permanent. The last tape used for the save is rewound at the end of the save operation. The set identifier associated with this save all set is ALLFL.

Example 4: Saving Multiple Files

```
SAVS36F   FROMFILE(*ALL)  FROMLIB(QS36F)  DEV(T1 T2)
          GROUP(*NONE)  CRTDATE(*LAST) SET(NOGFL) RETAIN(999)
```

This command saves all database physical and logical files in library QS36F except those files that belong to a file group. If any of the files are date-differentiated files, only the last member created in each file is saved. Assuming that T1 and T2 are tape devices, the files are copied to the tape volumes that are placed in tape drives T1 and T2. The label of the tape files created is the same as the names of the files that are saved. The first tape file created is located after the last sequence number on the tape. The remaining files are located after that first file. The tape files created are permanent. The last tape used for the save is rewound at the end of the save operation. The set identifier associated with this save all set is NOGFL.

Example 5: Saving Multiple Files

```
SAVS36F   FROMFILE(*ALL)  FROMLIB(GRPLIB)  DEV(I1)
          GROUP(GRP)  CRTDATE(*ALL)
```

This command saves all database physical and logical files in library GRPLIB that belong to file group GRP (GRP.01, GRP.02, and so on). If any of the files are date-differentiated files, all members in the files are saved. Assuming that I1 is a diskette drive, the files are copied to the diskette that is placed in the diskette drive. The label of the diskette files created is the same as the names of the files that are saved. The diskette files expire after one day. The set identifier associated with this save all set is #SAVE.

Example 6: Saving Multiple Files

```
SAVS36F   FROMFILE(PAY*)  FROMLIB(PAYROLL)  DEV(I1)
          SET(PAYSET)  CRTDATE(*LAST)
          VOL(PAYDKT)  RETAIN(10)
```

This command saves all database physical and logical files in library PAYROLL whose names begin with the characters PAY (PAY.01, PAYRATE, and so on). If any of the files are date-differentiated files, only the last member created is saved. Assuming that I1 is a diskette drive, the files are copied to a diskette with a volume identifier of PAYDKT. The label of the diskette files created is the same as the names of the files that are saved. The diskette files expire after ten days. The set identifier associated with this save all set is PAYSET.

Top

Error messages

***ESCAPE Messages**

CPF2C4A
Device &1 not correct for command.

CPF2C4B
Duplicate device &1 specified in device name list.

CPF2C4C
Diskette device &1 included in multiple device specification.

CPF2C4F
Diskette format not correct for DTACPR(*YES).

CPF2C47
Existing file &1 or member &3 in library &2 not replaced.

CPF2C48
Input file &1 in &2 not correct for command.

CPF2C49
Output file &1 in &2 not correct for command.

CPF2C5B
Not all files were saved.

CPF2C5C
Save operation ended before all files were saved.

CPF2C5D
No files saved from library &1.

CPF2C5E
Input file &1 in &2 not correct for command.

CPF2C5F
Tape file sequence numbers beyond 9999 not allowed.

CPF2C50
File description for file &1 is not available.

CPF2C51
Member information for file &1 in library &2 is not available.

CPF2C52
Error occurred during attempt to create file &1 in library &2.

CPF2C54
FROMFILE name &1 too long to use for TOLABEL parameter.

CPF2C55
TOLABEL parameter value &1 contains embedded blank(s).

CPF2C56
Physical file name &1 too long.

CPF2C58
Diskette format not acceptable for System/36.

CPF2C59
FROMFILE name &1 too long.

CPF9810
Library &1 not found.

CPF9812
File &1 in library &2 not found.

CPF9814
Device &1 not found.

CPF9820
Not authorized to use library &1.

CPF9822
Not authorized to file &1 in library &2.

CPF9825
Not authorized to device &1.

CPF9826
Cannot allocate file &2.

CPF9830
Cannot assign library &1.

CPF9831
Cannot assign device &1.

CPF9845
Error occurred while opening file &1.

CPF9847
Error occurred while closing file &1 in library &2.

CPF9848
Cannot open file &1 in library &2 member &3.

CPF9849
Error while processing file &1 in library &2 member &3.

*STATUS Messages

CPI2C13
Copying records from file &1 in library &2 member &3.

Top

Save S/36 Library Members (SAVS36LIBM)

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

Parameters
Examples
Error messages

The Save Library Members in System/36 Save/Restore Format (SAVS36LIBM) command creates a copy of source file members in a file on diskette or tape that can be restored on a System/36, or into a database physical file on this system that can be sent to a System/36. The saved member file is formatted like a record-mode LIBRFILE created on a System/36 using the System/36 FROMLIBR system operator control language (OCL) procedure (or the equivalent OCL use of the $MAINT SSP utility). On a System/36, the diskette or tape or disk file can be restored using the TOLIBR system OCL procedure (or the equivalent OCL use of the $MAINT SSP utility).

If a database physical file is specified using the **Physical file** prompt (PHYFILE parameter) but does not exist, it is created.

This command is intended only for exchanging source and procedure data with a System/36. It provides a simplified command interface for an i5/OS customer who migrated from a System/36, but is not well-suited for backing up of i5/OS source files. Use the CL commands (SAVOBJ or SAVCHGOBJ) for creating a backup copy of an i5/OS source file or individual source file members.

Restrictions:

1. The following authorities are required when running on a system using resource security:
   - *USE authority for this command
   - *USE authority for the library specified in the FROMLIB parameter
   - *USE authority for file QS36SRC in the specified library if saving source library members
   - *USE authority for file QS36PRC in the specified library if saving procedure library members
   - *USE authority for the library specified in the PHYFILE parameter if saving to a physical file
   - *CHANGE authority for the library specified in the PHYFILE parameter if saving to a physical file and the file does not exist
   - *CHANGE and *OBJMGMT authority for that file if saving to a physical file with MBROPT(*ADD)
   - *ALL authority for the file if saving to a physical file with MBROPT(*REPLACE)
   - *USE authority for the diskette device description object, *USE authority for device file QSYSDKT in library QSYS if saving to diskette
   - *USE authority for the tape device description object and *USE authority for device file QSYSTAP in library QSYS if saving to tape
2. All diskettes that are used to save the members should be initialized using the INZDKT CL command or the equivalent System/36 environment function (INIT OCL procedure or $INIT SSP utility). For a two-sided diskette, use a sector size of 256 or 1024. For a one-sided diskette, use a sector size of 128 or 512. If tape is used, each tape volume used should first be initialized with standard labels using the INZTAP CL command or the equivalent System/36 environment function (TAPEINIT OCL procedure or $TINIT SSP utility). Use a density of 1600 bits per inch when initializing the tape.

   Note: If the tape or diskette has not been initialized as stated above, the System/36 will not be able to process the media.

   If a tape or diskette is used that has not been properly initialized, a message is sent to the system operator that allows the operator to cancel the save or initialize the volume and continue.

3. Object-level functions, other than read operations, should not be used for files QS36SRC and QS36PRC while members are being saved by SAVS36LIBM. If the necessary files cannot be allocated, no members are saved.

   Record-level functions, other than read operations, should not be used for members being saved. Concurrent activity against a member (for example, adding or removing records) can cause the member to be omitted from the save operation.
4. If saving a file to diskette, the diskettes used cannot contain an active file with the same name as the TOLABEL parameter value, because the AS/400 system does not allow duplicate diskette file labels.
5. Only members from source files QS36SRC (for *SRC members) and QS36PRC (for *PRC members) in the specified library can be saved using the SAVS36LIBM command. Only the member data is saved from the source file member (that is, the source sequence number and change date fields are not saved).
6. The specified member name or generic member name (FROMMBR parameter) must meet i5/OS naming standards. When saving a member that has an extended name, the quotation mark characters are not stored as part of the member name in the output file. For example, member ″A+B″ would be saved as A+B.

Top

Parameters

Keyword	Description	Choices	Notes
FROMMBR	From member	*Generic name, name*, *ALL	Required, Positional 1
FROMLIB	From library	*Name*	Required, Positional 2
DEV	Device	Single values: *PHYFILE Other values (up to 4 repetitions): *Name*	Required, Positional 3
SRCMBRS	S/36 source members	***ALL**, *SRC, *PRC	Optional
TOLABEL	File label	*Character value*	Optional
SEQNBR	Sequence number	1-9999, ***END**	Optional
VOL	Volume identifier	Values (up to 50 repetitions): *Character value*, ***MOUNTED**	Optional
RETAIN	Retention period	0-999, **1**	Optional
EXCHTYPE	Diskette file exchange type	***E**, *BASIC	Optional
ENDOPT	End of tape option	***REWIND**, *LEAVE, *UNLOAD	Optional
PHYFILE	Physical file	*Qualified object name*	Optional
	Qualifier 1: Physical file	*Name*	
	Qualifier 2: Library	*Name*, *CURLIB, ***LIBL**	
MBROPT	Data base member option	***REPLACE**, *ADD	Optional
RCDLEN	Record length	40-120, **120**	Optional

Top

From member (FROMMBR)

Specifies the names of the members being saved.

This is a required parameter.

***ALL** All members of the member type specified on the **S/36 source members** prompt (SRCMBRS parameter) are saved.

member-name
Specify the member name of the members to be saved.

generic-member-name*
Specify the generic member name of the members to be saved. A generic name is a character string that contains one or more characters followed by an asterisk (*).

Top

From library (FROMLIB)

Specifies which library contains the members being saved.

This is a required parameter.

Top

Device (DEV)

Specifies the names of the devices used for the save operation. A maximum of four devices may be specified.

This is a required parameter.

***PHYFILE**
The output file is the database physical file specified by the **Physical file** prompt (PHYFILE parameter).

device-name
Specify the name of the diskette unit or the names of one or more tape devices that are used for the save operation. If more than one tape device is used, type the names of the devices in the order in which they are used. A maximum of four tape device names can be specified.

Top

S/36 source members (SRCMBRS)

Specifies which source member types (source and procedure members on System/36) are saved.

***ALL** System/36 source and procedure members (from QS36SRC and QS36PRC) that match the member name specified on the **From member** prompt (FROMMBR parameter) are saved.

***SRC** Only System/36 source members (from file QS36SRC) that match the member name specified on the **From member** prompt (FROMMBR parameter) are saved.

***PRC** Only System/36 OCL procedure members (from file QS36PRC) that match the member name specified on the **From member** prompt (FROMMBR parameter) are saved.

Top

File label (TOLABEL)

Specifies the label value (eight characters maximum) of the output diskette or tape file. If *PHYFILE is not specified on the **Device** prompt (DEV parameter), a value must be specified here.

Top

Sequence number (SEQNBR)

Specifies, only when tape is used, which sequence number is used for the save operation.

***END** The specified members are saved starting after the last sequence number on the first tape (this file is put after all other files currently on the tape volume). If the first tape is full, an error message is issued and the operation ends. If the sequence number to be assigned to the specified file is greater than 9999, an error message is issued and the operation ends.

file-sequence-number
Specify the sequence number of the tape file that is used when saving the specified members. Valid values range from 1 through 9999.

If this sequence number already exists on the tape volume, the tape label at that sequence number must match the TOLABEL parameter value. The existing data file at that sequence number is overwritten, and all subsequent files on the volume are not accessible after the save operation.

If a new tape file is added to the tape, the sequence number must be one greater than the sequence number of the last tape file on that volume. No gaps are allowed in the series of sequence numbers.

Top

Volume identifier (VOL)

Specifies the volume identifiers of the tape volumes on which the object data is to be saved. The volumes must be placed in the device in the same order as the volume identifiers are specified for this parameter.

***MOUNTED**
The volume currently placed in the device is used.

volume-identifier
Specify the volume identifiers of the tapes or diskettes used for saving the members. A maximum of 50 volume identifiers can be specified.

Top

Retention period (RETAIN)

Specifies the retention period for the newly created tape or diskette file. The file is protected and cannot be written over until the day after the retention period ends.

1 A retention period of one day is used.

retention-period
Specify the number of days the tape or diskette file should be kept. If a retention period of 999 is specified, the tape or diskette file becomes a permanent file.

Top

Diskette file exchange type (EXCHTYPE)

Specifies the exchange type for the newly created diskette file.

***E** The default is to create an E-exchange diskette file. An E-exchange file is a system save file that can be restored on a System/36 using the TOLIBR procedure. It can also be restored to the AS/400 system using the Restore System/36 Library Members (RSTS36LIBM) command.

***BASIC**
The output diskette file is to be in basic exchange format. A basic exchange format file can be restored or copied to a System/34 or System/32 or any other system that supports the basic exchange diskette format.

Top

End of tape option (ENDOPT)

Specifies, only when tape is used, what operation is automatically performed on the tape volume after the save operation ends. This parameter applies only to the last reel used.

***REWIND**
The tape is rewound, but not unloaded.

***LEAVE**
The tape is not rewound.

***UNLOAD**
The tape is automatically rewound and unloaded after the operation ends.

Top

Physical file (PHYFILE)

Specifies the name of the physical file that receives the copied source file member data.

If a file by this name does not exist, it is created in the current library if a library name was not specified, as a non-keyed, program-described physical file with the record length specified by the **Logical record length** prompt (RCDLEN parameter). If a file already exists by this name, it is used as long as it is a non-keyed physical file with a record length in the range from 40 through 120. The copied records are put in the first member of the physical 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, QGPL is used.

library-name
Specify the library where the file is located.

Top

Data base member option (MBROPT)

Specifies whether the new records replace or are added to the existing records.

***REPLACE**
The member is cleared before copying the first record.

***ADD** The system adds the new records to the end of the existing records.

Top

Record length (RCDLEN)

Specifies the file record length (in bytes) used when copying the members.

120 This is the maximum record length allowed for System/36 source and procedure library members.

record-length
Specify a record length in bytes.

Top

Examples

Example 1: Saving Single Procedure Member

```
SAVS36LIBM   FROMMBR(XYZ1)  FROMLIB(JOHNSON)
             DEV(I1)  SRCMBRS(*PRC)  TOLABEL(XYZ1)
```

This command saves the single OCL procedure member XYZ1 (in source file QS36PRC) in library JOHNSON. Assuming I1 is a diskette device, the member is saved to diskette file XYZ1. The file length is 120 and the retention period is one day.

Example 2: Saving All Source and Procedure Members

```
SAVS36LIBM   FROMMBR(X*)  FROMLIB(ORDER)  DEV(*PHYFILE)
             PHYFILE(NETLIB/S36SRC) MBROPT(*ADD)
```

This command saves all source and procedure members (members in QS36SRC and QS36PRC) with names starting with the character ’X’ in library ORDER. The members are saved into database physical file S36SRC in library NETLIB. The copied records are added after any records already in the file.

Top

Error messages

***ESCAPE Messages**

CPF2C4A
Device &1 not correct for command.

CPF2C4B
Duplicate device &1 specified in device name list.

CPF2C4C
Diskette device &1 included in multiple device specification.

CPF2C43
Saved &2 members from library &1, &3 members not saved.

CPF2C44
No members saved from library &1.

CPF2C48
Input file &1 in &2 not correct for command.

CPF2C49
Output file &1 in &2 not correct for command.

CPF2C5E
Input file &1 in &2 not correct for command.

CPF2C5F
Tape file sequence numbers beyond 9999 not allowed.

CPF2C50
File description for file &1 is not available.

CPF2C51
Member information for file &1 in library &2 is not available.

CPF2C52
Error occurred during attempt to create file &1 in library &2.

CPF2C55
TOLABEL parameter value &1 contains embedded blank(s).

CPF2C58
Diskette format not acceptable for System/36.

CPF9807
One or more libraries in library list deleted.

CPF9808
Cannot allocate one or more libraries on library list.

CPF9810
Library &1 not found.

CPF9812
File &1 in library &2 not found.

CPF9814
Device &1 not found.

CPF9820
Not authorized to use library &1.

CPF9822
Not authorized to file &1 in library &2.

CPF9825
Not authorized to device &1.

CPF9826
Cannot allocate file &2.

CPF9830
Cannot assign library &1.

CPF9845
Error occurred while opening file &1.

CPF9847
Error occurred while closing file &1 in library &2.

CPF9849
Error while processing file &1 in library &2 member &3.

*STATUS Messages

CPI2C13

Copying records from file &1 in library &2 member &3.

Top

Save Save File Data (SAVSAVFDTA)

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

Parameters
Examples
Error messages

The Save Save File Data (SAVSAVFDTA) command saves the contents of a save file to tape or optical media. This command saves the save data in the save file to the device in a way that allows you to restore objects directly from the device.

A save file containing data created by the Save Licensed Program (SAVLICPGM) command cannot be saved by this command.

The information written on media by this command is similar to the data that was previously written to the save file by the save command that originally created the save file data. This includes the object descriptions, and object contents that existed when the original save operation was done.

This command uses only the save file and device description objects; it does not refer to or modify the description or contents of the objects included in the file save data. Thus, objects included in the save file are not locked during the running of this command, and the save history information (date, place, and time when each object was last saved) is not updated by this command for each object in the save file.

The description of the save file is not included in the save operation (unless it was included with the objects that were saved to create the save data in the file). In addition, this command does not update the save history information for the save file object, so the last save operation date, time, and place always identify the last save operation of the save file object description, not its contents.

Note: This command ignores all file overrides currently in effect for the job, except for the output file.

Restrictions:

- You must have use (*USE) authority for the save file and *USE authority for the tape or optical device description.
- If a tape is used, it must have a standard label.
- The save file cannot be in use by a job running at the time the save operation occurs.

Top

Parameters

Keyword	Description	Choices	Notes
SAVF	Save file	*Qualified object name*	Required, Positional 1
	Qualifier 1: Save file	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
DEV	Device	Values (up to 4 repetitions): *Name*	Required, Positional 2
VOL	Volume identifier	Single values: ***MOUNTED** Other values (up to 75 repetitions): *Character value*	Optional, Positional 3
SEQNBR	Sequence number	1-16777215, ***END**	Optional
ENDOPT	End of media option	***REWIND**, *LEAVE, *UNLOAD	Optional
OPTFILE	Optical file	*Path name*, *****	Optional

Keyword	Description	Choices	Notes
USEOPTBLK	Use optimum block	***YES**, *NO	Optional
CLEAR	Clear	***NONE**, *ALL, *AFTER, *REPLACE	Optional
EXPDATE	File expiration date	*Date*, ***PERM**	Optional
COMPACT	Data compaction	***DEV**, *NO	Optional
OUTPUT	Output	***NONE**, *PRINT, *OUTFILE	Optional
OUTFILE	File to receive output	*Qualified object name*	Optional
	Qualifier 1: File to receive output	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
OUTMBR	Output member options	*Element list*	Optional
	Element 1: Member to receive output	*Name*, ***FIRST**	
	Element 2: Replace or add records	***REPLACE**, *ADD	

Top

Save file (SAVF)

Specifies the save file whose contents are to be saved.

Note: The save file must contain data written by a previous save command or an error message is sent, and its contents are not saved to media.

This is a required parameter.

Qualifier 1: Save file

name Specify the name of the save file whose contents are to be saved.

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 save file. If no current library entry exists in the library list, the QGPL library is used.

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

Top

Device (DEV)

Specifies the name of the device used for the save operation. The device name must already be known on the system by a device description.

This is a required parameter.

optical-device-name
Specify the name of the optical device used for the save operation.

tape-media-library-device-name
Specify the name of the tape media library device used for the save operation.

tape-device-name

Specify the names of one or more tape devices used for the save operation. If a virtual tape device is used, it must be the only device specified. If multiple tape devices are used, they must have compatible media formats and their names must be specified in the order in which they are used. Using more than one tape device permits one tape volume to be rewound and unloaded while another tape device processes the next tape volume.

Top

Volume identifier (VOL)

Specifies the volume identifiers of the volumes, or the cartridge identifiers of tapes in a tape media library device, on which the data is saved. The volumes must be placed in the device in the same order as specified for this parameter.

Single values

***MOUNTED**

The data is saved on the volumes placed in the device. For a media library device, the volume to be used is the next cartridge in the category mounted by the Set Tape Category (SETTAPCGY) command.

Note: This value cannot be specified when using an optical media library device.

Other values (up to 75 repetitions)

character-value

Specify the identifiers of one or more volumes in the order in which they are placed in a device and used to save the data.

Top

Sequence number (SEQNBR)

Specifies, when tape is used, the sequence number to use as the starting point for the save operation.

***END** The save operation begins after the last sequence number on the first tape. If the first tape is full, an error message is issued and the operation ends.

1-16777215

Specify the sequence number of the file to be used for the save operation.

Top

End of media option (ENDOPT)

Specifies the operation that is automatically done on the tape or optical volume after the save operation ends. If more than one volume is used, this parameter applies only to the last volume used; all other volumes are unloaded when the end of the volume is reached.

Note: This parameter is valid only if a tape or optical device name is specified for the DEV parameter. For optical devices, *UNLOAD is the only special value supported, *REWIND and *LEAVE will be ignored.

***REWIND**

The tape is automatically rewound, but not unloaded, after the operation has ended.

***LEAVE**
The tape does not rewind or unload after the operation ends. It remains at the current position on the tape drive.

***UNLOAD**
The tape is automatically rewound and unloaded after the operation ends. Some optical devices will eject the volume after the operation ends.

Top

Optical file (OPTFILE)

Specifies the path name of the optical file that is used for the save operation, beginning with the root directory of the volume.

For more information on specifying path names, refer to ″Object naming rules″ in the CL topic collection in the Programming category in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/.

***** The system generates an optical file name in the root directory of the optical volume.

'optical-directory-path-name/*'
The system generates an optical file name in the specified directory of the optical volume.

Top

Use optimum block (USEOPTBLK)

Specifies whether or not the optimum block size is used for the save operation.

Note: Specifying USEOPTBLK(*YES) may result in a tape that can be duplicated only to a device that supports the same block size.

***YES** The optimum block size supported by the device is used for Save commands. If the block size that is used is larger than a block size that is supported by all device types, then:

- Performance may improve.
- The tape file that is created is only compatible with a device that supports the block size used. Commands such as Duplicate Tape (DUPTAP) do not duplicate files unless the files are being duplicated to a device which supports the same block size that was used.
- The value for the DTACPR parameter is ignored.

***NO** The optimum block size supported by the device is not used. Save commands use the default block size supported by all device types. The tape volume can be duplicated to any media format using the Duplicate Tape (DUPTAP) command.

Top

Clear (CLEAR)

Specifies whether active data on the media is automatically cleared or replaced. Active data is any file on the media that has not expired. For saves to tape, clearing active data will make any files on the tape volume beyond the last file written by the save operation no longer accessible. For saves to optical, the files written by the save operation can be automatically replaced while other files on the volume remain active, or all active files can be automatically cleared. Clearing does not erase the data, it just makes the files no longer accessible.

Notes:

1. Clearing a tape does not initialize it. You should initialize tapes to a standard label format before the save command is issued by using the Initialize Tape (INZTAP) command and specifying a value for the NEWVOL parameter.
2. Clearing an optical volume does initialize it.
3. If a volume that is not initialized is encountered during the save operation, an inquiry message is sent and an operator can initialize the volume.

***NONE**
None of the media is automatically cleared. If the save operation encounters active data on a tape volume or in a save file, an inquiry message is sent, allowing the operator to either end the save operation or clear the media. If the save operation encounters the specified optical file, an inquiry message is sent, allowing the operator to either end the save operation or replace the file.

***ALL** All of the media is automatically cleared.

If tapes are used and a sequence number is specified for the SEQNBR parameter, the first tape is cleared beginning at that sequence number. All tapes following that first tape are completely cleared. To clear the entire first tape, SEQNBR(1) must be specified.

***AFTER**
All media after the first volume is automatically cleared. If the save operation encounters active data on the first tape volume, an inquiry message is sent, allowing the operator to either end the save operation or clear the media. If the save operation encounters the specified optical file on the first volume, an inquiry message is sent, allowing the operator to either end the save operation or replace the file.

***REPLACE**
Active data on the media is automatically replaced. Optical volumes are not initialized. Other media is automatically cleared in the same way as the *ALL value.

Top

File expiration date (EXPDATE)

Specifies the expiration date of the file created by the save operation. If a date is specified, the file is protected and cannot be overwritten until the specified expiration date.

Notes:

1. This parameter is valid for tape and optical files.
2. Specifying this parameter does not protect against a later save operation specifying CLEAR(*ALL).

***PERM**
The file is protected permanently.

date Specify the date when protection for the file ends.

Top

Data compaction (COMPACT)

Specifies whether device data compaction is performed.

***DEV** Device data compaction is performed if the data is saved to tape and all tape devices specified for the **Device (DEV)** parameter support the compaction feature.

***NO** Device data compaction is not performed.

Top

Output (OUTPUT)

Specifies whether a list with information about the saved objects is created. The information can be printed with the job's spooled output or directed to a database file.

***NONE**
No output listing is created.

***PRINT**
The output is printed with the job's spooled output.

***OUTFILE**
The output is directed to the database file specified for the **File to receive output (OUTFILE)** parameter.

Note: You must specify a database file name for the **File to receive output (OUTFILE)** parameter when OUTPUT(*OUTFILE) is specified.

Top

File to receive output (OUTFILE)

Specifies the database file to which the output of the command is directed. If the file does not exist, this command creates a database file in the specified library. If the file is created, the public authority for the file is the same as the create authority specified for the library in which the file is created. Use the Display Library Description (DSPLIBD) command to show the library's create authority.

Qualifier 1: File to receive output

name Specify the name of the database file to which the command output is directed.

Qualifier 2: Library

***LIBL** The library list is used to locate the file. If the file is not found, one is created in the current library. If no current library exists, the file will be created in the QGPL library.

***CURLIB**
The current library for the thread is used to locate the 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.

Note: If a new file is created, the system uses the IBM-supplied file QASAVOBJ with format name QSRSAV as a model.

Top

Output member options (OUTMBR)

Specifies the name of the database file member to which the output is directed when *OUTFILE is specified for the **Output (OUTPUT)** parameter.

Element 1: Member to receive output

***FIRST**
The first member in the file receives the output. If OUTMBR(*FIRST) is specified and the member does not exist, the system creates a member with the name of the file specified for the **File to receive output (OUTFILE)** parameter.

name Specify the name of the file member that receives the output. If OUTMBR(member-name) is specified and the member does not exist, the system creates it.

If the member exists, you can add records to the end of the existing member or clear the existing member and add the records.

Element 2: Replace or add records

***REPLACE**
The existing records in the specified database file member are replaced by the new records.

***ADD** The new records are added to the existing information in the specified database file member.

Top

Examples

```
SAVSAVFDTA   SAVF(ONLINE)  DEV(TAP01)  SEQNBR(1)  CLEAR(*ALL)
```

This command saves the contents of save file ONLINE to the first file on the tape volume on device TAP01. Files that have not ended on either the first tape volume or on subsequent volumes are overwritten without an inquiry message because CLEAR(*ALL) is specified.

Top

Error messages

***ESCAPE Messages**

CPF3707
Save file &1 in &2 contains no data.

CPF3709
Tape devices do not support same densities.

CPF3727
Duplicate device &1 specified on device name list.

CPF3728
Device &1 specified with other devices.

CPF3733
&2 &1 in &3 previously damaged.

CPF3767
Device &1 not found.

CPF3768
Device &1 not valid for command.

CPF3782
File &1 in &2 not a save file.

CPF3793
Machine or ASP storage limit reached.

CPF3794
Save or restore operation ended unsuccessfully.

CPF3805
Objects from save file &1 in &2 not restored.

CPF3812
Save file &1 in &2 in use.

CPF3814
No objects from save file &1 in &2 saved to media.

CPF384E
USEOPTBLK(*YES) not valid for CD-ROM premastering.

CPF388B
Optical file path name not valid.

CPF5729
Not able to allocate object &1.

CPF9812
File &1 in library &2 not found.

CPF9822
Not authorized to file &1 in library &2.

CPF9825
Not authorized to device &1.

Top

Save Security Data (SAVSECDTA)

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

Parameters
Examples
Error messages

The Save Security Data (SAVSECDTA) command saves all security information without requiring a system in a restricted state. The SAVSECDTA command saves the same security information that is saved when a Save System (SAVSYS) command is run including the following:

- User Profiles
- Authorization Lists
- Authority Holders

Information saved with the SAVSYS or SAVSECDTA command can be restored using the Restore User Profiles (RSTUSRPRF) and Restore Authority (RSTAUT) commands.

Restrictions:

- You must have save system (*SAVSYS) special authority to run this command.
- Changes made to user profiles while the SAVSECDTA command is being run may not be reflected on the media, depending on when the changes occurred in relation to the save operation.
- Concurrent running of other SAVSECDTA commands is not allowed.
- If *YES is specified for the **Object pre-check (PRECHK)** parameter and a security object cannot be saved, the save operation ends.

Top

Parameters

Keyword	Description	Choices	Notes
DEV	Device	Single values: *SAVF Other values (up to 4 repetitions): *Name*	Required, Positional 1
VOL	Volume identifier	Single values: ***MOUNTED** Other values (up to 75 repetitions): *Character value*	Optional
SEQNBR	Sequence number	1-16777215, ***END**	Optional
EXPDATE	File expiration date	*Date*, ***PERM**	Optional
ENDOPT	End of media option	***REWIND**, *LEAVE, *UNLOAD	Optional
USEOPTBLK	Use optimum block	***YES**, *NO	Optional
SAVF	Save file	*Qualified object name*	Optional
	Qualifier 1: Save file	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
OPTFILE	Optical file	*Path name*, *****	Optional
ASPDEV	ASP device	*Name*, ***ALLAVL**, *, *SYSBAS, *CURASPGRP	Optional
CLEAR	Clear	***NONE**, *ALL, *AFTER, *REPLACE	Optional
PRECHK	Object pre-check	***NO**, *YES	Optional
DTACPR	Data compression	***DEV**, *NO, *YES, *LOW, *MEDIUM, *HIGH	Optional
COMPACT	Data compaction	***DEV**, *NO	Optional
OUTPUT	Output	***NONE**, *PRINT, *OUTFILE	Optional

Keyword	Description	Choices	Notes
OUTFILE	File to receive output	*Qualified object name*	Optional
	Qualifier 1: File to receive output	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
OUTMBR	Output member options	*Element list*	Optional
	Element 1: Member to receive output	*Name*, ***FIRST**	
	Element 2: Replace or add records	***REPLACE**, *ADD	

Top

Device (DEV)

Specifies the name of the device used for the save operation. The device name must already be known on the system by a device description.

This is a required parameter.

Single values

***SAVF** The save operation is done using the save file specified for the **Save file (SAVF)** parameter.

Other values

optical-device-name
Specify the name of the optical device used for the save operation.

tape-media-library-device-name
Specify the name of the tape media library device used for the save operation.

tape-device-name
Specify the names of one or more tape devices used for the save operation. If a virtual tape device is used, it must be the only device specified. If multiple tape devices are used, they must have compatible media formats and their names must be specified in the order in which they are used. Using more than one tape device permits one tape volume to be rewound and unloaded while another tape device processes the next tape volume.

Top

Volume identifier (VOL)

Specifies the volume identifiers of the volumes, or the cartridge identifiers of tapes in a tape media library device, on which the data is saved. The volumes must be placed in the device in the same order as specified for this parameter.

Single values

***MOUNTED**
The data is saved on the volumes placed in the device. For a media library device, the volume to be used is the next cartridge in the category mounted by the Set Tape Category (SETTAPCGY) command.

Note: This value cannot be specified when using an optical media library device.

Other values (up to 75 repetitions)

character-value
Specify the identifiers of one or more volumes in the order in which they are placed in a device and used to save the data.

Top

Sequence number (SEQNBR)

Specifies, when tape is used, the sequence number to use as the starting point for the save operation.

***END** The save operation begins after the last sequence number on the first tape. If the first tape is full, an error message is issued and the operation ends.

1-16777215
Specify the sequence number of the file to be used for the save operation.

Top

File expiration date (EXPDATE)

Specifies the expiration date of the file created by the save operation. If a date is specified, the file is protected and cannot be overwritten until the specified expiration date.

Notes:

1. This parameter is valid for tape and optical files.
2. Specifying this parameter does not protect against a later save operation specifying CLEAR(*ALL).

***PERM**
The file is protected permanently.

date Specify the date when protection for the file ends.

Top

End of media option (ENDOPT)

Specifies the operation that is automatically done on the tape or optical volume after the save operation ends. If more than one volume is used, this parameter applies only to the last volume used; all other volumes are unloaded when the end of the volume is reached.

Note: This parameter is valid only if a tape or optical device name is specified for the DEV parameter. For optical devices, *UNLOAD is the only special value supported, *REWIND and *LEAVE will be ignored.

***REWIND**
The tape is automatically rewound, but not unloaded, after the operation has ended.

***LEAVE**
The tape does not rewind or unload after the operation ends. It remains at the current position on the tape drive.

***UNLOAD**
The tape is automatically rewound and unloaded after the operation ends. Some optical devices will eject the volume after the operation ends.

Top

Use optimum block (USEOPTBLK)

Specifies whether or not the optimum block size is used for the save operation.

Note: Specifying USEOPTBLK(*YES) may result in a tape that can be duplicated only to a device that supports the same block size.

<u>*YES</u> The optimum block size supported by the device is used for Save commands. If the block size that is used is larger than a block size that is supported by all device types, then:

- Performance may improve.
- The tape file that is created is only compatible with a device that supports the block size used. Commands such as Duplicate Tape (DUPTAP) do not duplicate files unless the files are being duplicated to a device which supports the same block size that was used.
- The value for the DTACPR parameter is ignored.

***NO** The optimum block size supported by the device is not used. Save commands use the default block size supported by all device types. The tape volume can be duplicated to any media format using the Duplicate Tape (DUPTAP) command.

Top

Save file (SAVF)

Specifies the save file that is used to contain the saved data. The save file must be empty, unless *ALL is specified for the **Clear (CLEAR)** parameter.

Note: A value must be specified for this parameter if *SAVF is specified for the **Device (DEV)** parameter.

Qualifier 1: Save file

name Specify the name of save file to be used.

Qualifier 2: Library

<u>*LIBL</u> 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 save file. If no current library entry exists in the library list, the QGPL library is used.

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

Top

Optical file (OPTFILE)

Specifies the path name of the optical file that is used for the save operation, beginning with the root directory of the volume.

For more information on specifying path names, refer to ″Object naming rules″ in the CL topic collection in the Programming category in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/.

<u>*</u> The system generates an optical file name in the root directory of the optical volume.

'optical-directory-path-name/*'
The system generates an optical file name in the specified directory of the optical volume.

Top

ASP device (ASPDEV)

Specifies the auxiliary storage pool (ASP) device from which private authorities are to be saved.

***ALLAVL**
The private authorities from the system ASP (ASP number 1), all basic user ASPs (ASP numbers 2-32), and all available independent ASPs are saved.

* The private authorities from the system ASP, all basic user ASPs, and, if the current thread has an ASP group, all independent ASPs in the ASP group are saved.

***SYSBAS**
The private authorities from the system ASP and all basic user ASPs are saved.

***CURASPGRP**
If the current thread has an ASP group, the private authorities from all independent ASPs in the ASP group are saved.

name Specify the ASP device name from which private authorities are to be saved.

Top

Clear (CLEAR)

Specifies whether active data on the media is automatically cleared or replaced. Active data is any file on the media that has not expired. For saves to tape, clearing active data will make any files on the tape volume beyond the last file written by the save operation no longer accessible. For saves to optical, the files written by the save operation can be automatically replaced while other files on the volume remain active, or all active files can be automatically cleared. Clearing does not erase the data, it just makes the files no longer accessible.

Notes:

1. Clearing a tape does not initialize it. You should initialize tapes to a standard label format before the save command is issued by using the Initialize Tape (INZTAP) command and specifying a value for the NEWVOL parameter.
2. Clearing an optical volume does initialize it.
3. If a volume that is not initialized is encountered during the save operation, an inquiry message is sent and an operator can initialize the volume.

***NONE**
None of the media is automatically cleared. If the save operation encounters active data on a tape volume or in a save file, an inquiry message is sent, allowing the operator to either end the save operation or clear the media. If the save operation encounters the specified optical file, an inquiry message is sent, allowing the operator to either end the save operation or replace the file.

***ALL** All of the media is automatically cleared.

If tapes are used and a sequence number is specified for the SEQNBR parameter, the first tape is cleared beginning at that sequence number. All tapes following that first tape are completely cleared. To clear the entire first tape, SEQNBR(1) must be specified.

***AFTER**
All media after the first volume is automatically cleared. If the save operation encounters active

data on the first tape volume, an inquiry message is sent, allowing the operator to either end the save operation or clear the media. If the save operation encounters the specified optical file on the first volume, an inquiry message is sent, allowing the operator to either end the save operation or replace the file.

Note: The *AFTER value is not valid for save files.

***REPLACE**
Active data on the media is automatically replaced. Optical volumes are not initialized. Other media is automatically cleared in the same way as the *ALL value.

Top

Object pre-check (PRECHK)

Specifies whether the save operation should end if all objects specified by this command do not satisfy the following conditions of the save operation: (1) the objects exist, (2) they were not previously found to be damaged, (3) they are not locked by another job, and (4) the requester of the save operation has authority to save the objects.

***NO** The save operation continues, saving only those objects that can be saved.

***YES** If, after all specified objects are checked, one or more objects cannot be saved, the save operation ends before any data is written.

Top

Data compression (DTACPR)

Specifies whether data compression is used. If the save is running while other jobs on the system are active and software compression is used, the overall system performance may be affected.

Note: If *DEV is specified for both this parameter and the **Data compaction (COMPACT)** parameter, only device data compaction is performed if device data compaction is supported on the device. Otherwise, data compression is performed.

If *YES is specified for this parameter and *DEV is specified for the COMPACT parameter, both device data compaction and device data compression are performed if supported on the device.

***DEV** If the save is to tape and the target device supports compression, hardware compression is performed. Otherwise, no data compression is performed.

***NO** No data compression is performed.

***YES** If the save is to tape and the target device supports compression, hardware compression is performed. If compression is not supported, or if the save data is written to optical media or to a save file, software compression is performed. Low software compression is used for all devices except optical DVD, which uses medium software compression.

***LOW** If the save operation is to a save file or optical, software data compression is performed with the SNA algorithm. Low compression is usually faster and the compressed data is usually larger than if medium or high compression is used.

Note: This value is not valid for tape.

***MEDIUM**
If the save operation is to a save file or optical, software data compression is performed with the

TERSE algorithm. Medium compression is usually slower than low compression but faster than high compression. The compressed data is usually smaller than if low compression is used and larger than if high compression is used.

Note: This value is not valid for tape.

***HIGH**
If the save operation is to a save file or optical, software data compression is performed with the LZ1 algorithm. High compression is usually slower and the compressed data is usually smaller than if low or medium compression is used.

Note: This value is not valid for tape.

Top

Data compaction (COMPACT)

Specifies whether device data compaction is performed.

***DEV** Device data compaction is performed if the data is saved to tape and all tape devices specified for the **Device (DEV)** parameter support the compaction feature.

Note: If *DEV is specified for both the **Data compression (DTACPR)** parameter and this parameter, only device data compaction is performed if device data compaction is supported on the device. Otherwise, data compression is performed if supported on the device.

If *YES is specified for the DTACPR parameter and *DEV is specified for this parameter, both device data compaction and device data compression are performed if supported on the device.

***NO** Device data compaction is not performed.

Top

Output (OUTPUT)

Specifies whether a list with information about the saved objects is created. The information can be printed with the job's spooled output or directed to a database file.

***NONE**
No output listing is created.

***PRINT**
The output is printed with the job's spooled output.

***OUTFILE**
The output is directed to the database file specified for the **File to receive output (OUTFILE)** parameter.

Note: You must specify a database file name for the **File to receive output (OUTFILE)** parameter when OUTPUT(*OUTFILE) is specified.

Top

File to receive output (OUTFILE)

Specifies the database file to which the output of the command is directed. If the file does not exist, this command creates a database file in the specified library. If the file is created, the public authority for the file is the same as the create authority specified for the library in which the file is created. Use the Display Library Description (DSPLIBD) command to show the library's create authority.

Qualifier 1: File to receive output

name Specify the name of the database file to which the command output is directed.

Qualifier 2: Library

<u>*LIBL</u> The library list is used to locate the file. If the file is not found, one is created in the current library. If no current library exists, the file will be created in the QGPL library.

***CURLIB**
The current library for the thread is used to locate the 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.

Note: If a new file is created, the system uses the IBM-supplied file QASAVOBJ with format name QSRSAV as a model.

Top

Output member options (OUTMBR)

Specifies the name of the database file member to which the output is directed when *OUTFILE is specified for the **Output (OUTPUT)** parameter.

Element 1: Member to receive output

<u>*FIRST</u>
The first member in the file receives the output. If OUTMBR(*FIRST) is specified and the member does not exist, the system creates a member with the name of the file specified for the **File to receive output (OUTFILE)** parameter.

name Specify the name of the file member that receives the output. If OUTMBR(member-name) is specified and the member does not exist, the system creates it.

If the member exists, you can add records to the end of the existing member or clear the existing member and add the records.

Element 2: Replace or add records

<u>*REPLACE</u>
The existing records in the specified database file member are replaced by the new records.

***ADD** The new records are added to the existing information in the specified database file member.

Top

Examples

Example 1: Automatically Clearing Uncleared Tapes

```
SAVSECDTA   DEV(TAP01)  CLEAR(*ALL)
```

This command saves the security information, including user profiles, authorization lists, authority holders. They are saved on the TAP01 tape drive. CLEAR(*ALL) automatically clears all uncleared tapes when they are encountered.

Example 2: Sending Message When Storage Capacity Exceeded

```
SAVSECDTA   DEV(TAP01)  VOL(ABC)
```

This command saves the security information on the TAP01 tape drive, starting on the tape volume labeled ABC. If the save operation exceeds the storage capacity of one tape, a message requesting that another volume be put on the TAP01 tape drive is shown to the operator.

Top

Error messages

***ESCAPE Messages**

CPF2206
User needs authority to do requested function on object.

CPF222E
&1 special authority is required.

CPF370A
Not all security objects saved to save file &3.

CPF3709
Tape devices do not support same densities.

CPF3727
Duplicate device &1 specified on device name list.

CPF3728
Device &1 specified with other devices.

CPF3731
Cannot use &2 &1 in library &3.

CPF3733
&2 &1 in &3 previously damaged.

CPF3735
Storage limit exceeded for user profile &1.

CPF3737
Save and restore data area &1 not found.

CPF3738
Device &1 used for save or restore is damaged.

CPF3767
Device &1 not found.

CPF3768
Device &1 not valid for command.

CPF3782
File &1 in &2 not a save file.

CPF3793
Machine or ASP storage limit reached.

CPF3794
Save or restore operation ended unsuccessfully.

CPF3812
Save file &1 in &2 in use.

CPF384E
USEOPTBLK(*YES) not valid for CD-ROM premastering.

CPF388B
Optical file path name not valid.

CPF3893
Not all security objects saved.

CPF3894
Cancel reply received for message &1.

CPF38A4
ASP device &1 not correct.

CPF5729
Not able to allocate object &1.

CPF9809
Library &1 cannot be accessed.

CPF9812
File &1 in library &2 not found.

CPF9814
Device &1 not found.

CPF9833
*CURASPGRP or *ASPGRPPRI specified and thread has no ASP group.

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.

CPF9850
Override of printer file &1 not allowed.

CPF9851
Overflow value for file &1 in &2 too small.

CPF9860
Error occurred during output file processing.

CPFB8ED
Device description &1 not correct for operation.

Top

Save Storage (SAVSTG)

Where allowed to run: Interactive environments (*INTERACT *IPGM *IREXX *EXEC)
Threadsafe: No

Parameters
Examples
Error messages

The Save Storage (SAVSTG) command saves a copy of the licensed internal code and the contents of auxiliary storage (except unused space and temporary objects) to tape. This function is intended for disaster recovery backup. Individual libraries or objects cannot be restored from a save storage tape.

This command issues the PWRDWNSYS (Power Down System) command with OPTION(*IMMED) and RESTART(*YES) specified. While the system is powered down, a dedicated service tool (DST) that saves all system storage is called. At that time, a standard labeled tape volume must be placed in the tape device. Additional volumes are requested as needed. Hardware data compression is used if it is supported by the tape device. After the save system storage function is complete, an initial program load (IPL) takes place.

During the IPL after a save storage operation, a completion message is sent to the system operator message queue. The save history information for the data area QSAVSTG in library QSYS is updated with the date and time when the system storage data was saved. To show the information in this data area, use the Display Object Description (DSPOBJD) command with DETAIL(*FULL).

The restore storage operation is done using the appropriate option on the DST menu. During the IPL, after a restore storage operation, a completion message is sent to the system operator message queue, and the last restore date and time history information in the QSAVSTG data area is updated with the current date and time. In addition, the data portion of the QSAVSTG data area is updated with the date of the save storage tape used in the restore system storage operation.

Note: Because media errors cause the save operation to start over from the last tape volume, use of this command is recommended for smaller systems only.

Restrictions:

- You must have save system (*SAVSYS) special authority to run this command.
- The system cannot be running any other jobs; run the End Subsystem (ENDSBS) specifying SBS(*ALL) or run the End System (ENDSYS) command to end all other jobs except the job running the SAVSTG command.
- Tapes created using this command that will be used for installation should be initialized with a density that is supported by the current alternate IPL device. If this is not done, the current IPL tape will have to be changed to a tape device that supports the density of the created SAVSTG tapes before installation can begin.
- Tapes created using the SAVSTG command should not be used for automatic installation.

Top

Parameters

Keyword	Description	Choices	Notes
DEV	Tape device	*Name*	Required, Positional 1
EXPDATE	File expiration date	*Date*, ***PERM**	Optional

Keyword	Description	Choices	Notes
CLEAR	Clear	***NONE**, *ALL	Optional

Top

Tape device (DEV)

Specifies the tape device that is used for the save storage operation.

This is a required parameter.

name Specify the name of the tape device to be used.

Top

File expiration date (EXPDATE)

Specifies the expiration date of the file created by the save operation. If a date is specified, the file is protected and cannot be overwritten until the specified expiration date.

***PERM**
The tape files are protected permanently.

date Specify the date when protection for the tape files ends. The date must be specified in the job date format.

Top

Clear (CLEAR)

Specifies whether uncleared tapes encountered during the save operation are automatically cleared. An uncleared tape is one containing a file with an expiration date later than the date of the save operation, which includes files protected permanently using EXPDATE(*PERM).

Note: This parameter does not control initializing the tapes. The tapes used to perform the save operation should be initialized to a standard label format before the save command is issued. You can use the Initialize Tape (INZTAP) command and specify a value for the NEWVOL parameter to initialize a tape to a standard label format. If a tape volume that is not initialized is encountered during the save operation, an inquiry message is sent and an operator can initialize the tape volume.

***NONE**
None of the uncleared tapes encountered during the save operation are automatically cleared. If the save operation cannot proceed because an uncleared tape is encountered, an inquiry message is sent to the operator, who is allowed to end the save operation or to specify that the selected tape be cleared so the operation can continue.

***ALL** All uncleared tapes encountered during the save operation are automatically cleared.

Top

Examples

Example 1: Specifying Expiration Date

```
SAVSTG   DEV(TAP01)  EXPDATE(122290)  CLEAR(*ALL)
```

This command saves the system storage on the tape put on the TAP01 tape drive. Each uncleared tape is cleared automatically. The tape files written are protected and cannot be overwritten until December 22, 1990.

Example 2: Saving System Storage

```
SAVSTG   DEV(TAP02)
```

The system storage is saved on tape drive TAP02. CLEAR was not specified, so uncleared tapes encountered during the save operation cause an inquiry message to be sent to the operator, who either ends the save operation or specifies that the currently selected tape be cleared so the operation can continue. Because EXPDATE also was not specified, the tape files being written are protected permanently.

Top

Error messages

***ESCAPE Messages**

CPF2206
User needs authority to do requested function on object.

CPF222E
&1 special authority is required.

CPF376A
System must not be in manual mode.

CPF3767
Device &1 not found.

CPF3768
Device &1 not valid for command.

CPF3785
Not all subsystems ended.

Top

Save System (SAVSYS)

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

Parameters
Examples
Error messages

The Save System (SAVSYS) command saves a copy of the Licensed Internal Code and the QSYS library in a format compatible with the installation of the operating system. It does not save objects from any other library. In addition, it saves security and configuration objects that can also be saved using the Save Security Data (SAVSECDTA) and Save Configuration (SAVCFG) commands.

To save the system data on offline storage, the system writes a copy of the objects onto the media. The libraries and objects are not affected on the system. This command cannot be used to free any space occupied by these objects. The history information for the data area QSAVUSRPRF in QSYS is updated with the date, time, and place where user profiles are saved. The history information for the data area QSAVSYS in QSYS is updated with the date, time, and place where the system is saved. The history information for the data area QSAVCFG in QSYS is updated with the date, time, and place where configuration objects are saved. The history information is not updated for the individual objects. To display the information in these data areas, run the Display Object Description (DSPOBJD) command, and specify DETAIL(*FULL). Save the information from the display of QSAVUSRPRF for the location where the user profiles are saved.

When using this command, it is important to use the device on the system that is defined as the initial program load (IPL) device. The IPL device was defined by the service representative when the system was installed. If an IPL device is not used when using this command, then the system cannot be restored using the SAVSYS media (if densities or media types are incompatible).

Restrictions:

1. You must have save system (*SAVSYS) special authority to run this command
2. All subsystems must be inactive before the SAVSYS command can be specified. The End System (ENDSYS) or End Subsystem (ENDSBS) command can be used to make the subsystems inactive. You must have job control (*JOBCTL) special authority to use the ENDSYS or the ENDSBS command.
3. Tapes created using this command that will be used for installation should be initialized with a density that is supported by the current IPL tape unit. If this is not done, the current IPL tape will have to be changed to a tape device that supports the density of the created SAVSYS tapes before installation begins.
4. Tapes created using the SAVSYS command should not be used for automatic installation.

Top

Parameters

Keyword	Description	Choices	Notes
DEV	Device	Values (up to 4 repetitions): *Name*	Required, Positional 1
VOL	Volume identifier	Single values: ***MOUNTED** Other values (up to 75 repetitions): *Character value*	Optional, Positional 2
EXPDATE	File expiration date	*Date*, ***PERM**	Optional
ENDOPT	End of media option	***REWIND**, *LEAVE, *UNLOAD	Optional
USEOPTBLK	Use optimum block	***YES**, *NO	Optional

Keyword	Description	Choices	Notes
OMIT	Omit	Single values: ***NONE** Other values (up to 2 repetitions): *CFG, *SECDTA	Optional
OUTPUT	Output	***NONE**, *PRINT, *OUTFILE	Optional
OUTFILE	File to receive output	*Qualified object name*	Optional
	Qualifier 1: File to receive output	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
OUTMBR	Output member options	*Element list*	Optional
	Element 1: Member to receive output	*Name*, ***FIRST**	
	Element 2: Replace or add records	***REPLACE**, *ADD	
ASPDEV	ASP device	*Name*, ***ALLAVL**, *, *SYSBAS, *CURASPGRP	Optional
CLEAR	Clear	***NONE**, *ALL, *AFTER, *REPLACE	Optional
DTACPR	Data compression	***DEV**, *NO, *YES, *LOW, *MEDIUM, *HIGH	Optional
COMPACT	Data compaction	***DEV**, *NO	Optional

Top

Device (DEV)

Specifies the name of the device used for the save operation. The device name must already be known on the system by a device description.

This is a required parameter.

optical-device-name
Specify the name of the optical device used for the save operation.

tape-media-library-device-name
Specify the name of the tape media library device used for the save operation.

tape-device-name
Specify the names of one or more tape devices used for the save operation. If a virtual tape device is used, it must be the only device specified. If multiple tape devices are used, they must have compatible media formats and their names must be specified in the order in which they are used. Using more than one tape device permits one tape volume to be rewound and unloaded while another tape device processes the next tape volume.

Top

Volume identifier (VOL)

Specifies the volume identifiers of the volumes, or the cartridge identifiers of tapes in a tape media library device, on which the data is saved. The volumes must be placed in the device in the same order as specified for this parameter.

Single values

***MOUNTED**
The data is saved on the volumes placed in the device. For a media library device, the volume to be used is the next cartridge in the category mounted by the Set Tape Category (SETTAPCGY) command.

Note: This value cannot be specified when using an optical media library device.

Other values (up to 75 repetitions)

character-value
Specify the identifiers of one or more volumes in the order in which they are placed in a device and used to save the data.

Top

File expiration date (EXPDATE)

Specifies the expiration date of the file created by the save operation. If a date is specified, the file is protected and cannot be overwritten until the specified expiration date.

***PERM**
The files are protected permanently.

date Specify the date when protection for the files ends.

Top

End of media option (ENDOPT)

Specifies the operation that is automatically done on the tape or optical volume after the save operation ends. If more than one volume is used, this parameter applies only to the last volume used; all other volumes are unloaded when the end of the volume is reached.

Note: This parameter is valid only if a tape or optical device name is specified for the DEV parameter. For optical devices, *UNLOAD is the only special value supported, *REWIND and *LEAVE will be ignored.

***REWIND**
The tape is automatically rewound, but not unloaded, after the operation has ended.

***LEAVE**
The tape does not rewind or unload after the operation ends. It remains at the current position on the tape drive.

***UNLOAD**
The tape is automatically rewound and unloaded after the operation ends. Some optical devices will eject the volume after the operation ends.

Top

Use optimum block (USEOPTBLK)

Specifies whether or not the optimum block size is used for the save operation.

Note: Specifying USEOPTBLK(*YES) may result in a tape that can be duplicated only to a device that supports the same block size.

***YES** The optimum block size supported by the device is used for Save commands. If the block size that is used is larger than a block size that is supported by all device types, then:

- Performance may improve.

- The tape file that is created is only compatible with a device that supports the block size used. Commands such as Duplicate Tape (DUPTAP) do not duplicate files unless the files are being duplicated to a device which supports the same block size that was used.
- The value for the DTACPR parameter is ignored.

***NO** The optimum block size supported by the device is not used. Save commands use the default block size supported by all device types. The tape volume can be duplicated to any media format using the Duplicate Tape (DUPTAP) command.

Note: Not all files will be saved with optimum block size.

Top

Omit (OMIT)

Specifies what data to omit from the save system operation.

Single values

***NONE**
All of the security data and configuration data are included in the save system operation.

Other values (up to 2 repetitions)

***CFG** All of the configuration data is omitted from the save system operation. You can use the SAVCFG (Save Configuration) command to save just the configuration object information.

***SECDTA**
All of the security data is omitted from the save system operation. You can use the SAVSECDTA (Save Security Data) command to save just the system security data.

Top

Output (OUTPUT)

Specifies whether a list with information about the saved objects is created. The information can be printed with the job's spooled output or directed to a database file.

***NONE**
No output listing is created.

***PRINT**
The output is printed with the job's spooled output.

***OUTFILE**
The output is directed to the database file specified for the **File to receive output (OUTFILE)** parameter.

Note: You must specify a database file name for the **File to receive output (OUTFILE)** parameter when OUTPUT(*OUTFILE) is specified.

Top

File to receive output (OUTFILE)

Specifies the database file to which the output of the command is directed. If the file does not exist, this command creates a database file in the specified library. If the file is created, the public authority for the file is the same as the create authority specified for the library in which the file is created. Use the Display Library Description (DSPLIBD) command to show the library's create authority.

Qualifier 1: File to receive output

name Specify the name of the database file to which the command output is directed.

Qualifier 2: Library

***LIBL** The library list is used to locate the file. If the file is not found, one is created in the current library. If no current library exists, the file will be created in the QGPL library.

***CURLIB**
The current library for the thread is used to locate the 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.

Note: If a new file is created, the system uses the IBM-supplied file QASAVOBJ with format name QSRSAV as a model.

Top

Output member options (OUTMBR)

Specifies the name of the database file member to which the output is directed when *OUTFILE is specified for the **Output (OUTPUT)** parameter.

Element 1: Member to receive output

***FIRST**
The first member in the file receives the output. If OUTMBR(*FIRST) is specified and the member does not exist, the system creates a member with the name of the file specified for the **File to receive output (OUTFILE)** parameter.

name Specify the name of the file member that receives the output. If OUTMBR(member-name) is specified and the member does not exist, the system creates it.

If the member exists, you can add records to the end of the existing member or clear the existing member and add the records.

Element 2: Replace or add records

***REPLACE**
The existing records in the specified database file member are replaced by the new records.

***ADD** The new records are added to the existing information in the specified database file member.

Top

ASP device (ASPDEV)

Specifies the auxiliary storage pool (ASP) device from which private authorities are to be saved.

***ALLAVL**
The private authorities from the system ASP (ASP number 1), all basic user ASPs (ASP numbers 2-32), and all available independent ASPs are saved.

* The private authorities from the system ASP, all basic user ASPs, and, if the current thread has an ASP group, all independent ASPs in the ASP group are saved.

***SYSBAS**
The private authorities from the system ASP and all basic user ASPs are saved.

***CURASPGRP**
If the current thread has an ASP group, the private authorities from all independent ASPs in the ASP group are saved.

name Specify the ASP device name from which private authorities are to be saved.

Top

Clear (CLEAR)

Specifies whether active data on the media is automatically cleared or replaced. Active data is any file on the media that has not expired. Clearing active data removes all files from the volume. Replacing active data on optical media initializes the first volume, and replaces only the optical files created by this operation on any volumes after the first volume.

Notes:

1. Clearing a tape does not initialize it. You should initialize tapes to a standard label format before the save command is issued by using the Initialize Tape (INZTAP) command and specifying a value for the NEWVOL parameter.
2. Clearing an optical volume does initialize it.
3. If a volume that is not initialized is encountered during the save operation, an inquiry message is sent and an operator can initialize the volume.

***NONE**
None of the media is automatically cleared. If the save operation encounters active data on the first volume, an inquiry message is sent, allowing the operator to either end the save operation or clear the media. If the save operation encounters active data after the first volume, an inquiry message is sent, allowing the operator to either end the save operation, clear the tape, or replace the optical file.

***ALL** All of the media is automatically cleared.

***AFTER**
All media after the first volume is automatically cleared. If the save operation encounters active data on the first tape, an inquiry message is sent, allowing the operator to either end the save operation or clear the media. If the save operation encounters the specified optical file on the first volume, an inquiry message is sent, allowing the operator to either end the save operation or replace the file.

***REPLACE**
Active data on the media is automatically replaced. The first optical volume is initialized. Other optical volumes are not initialized. Tapes are automatically cleared in the same way as the *ALL value.

Top

Data compression (DTACPR)

Specifies whether data compression is used. If the save is running while other jobs on the system are active and software compression is used, the overall system performance may be affected.

Note: If *DEV is specified for both this parameter and the **Data compaction (COMPACT)** parameter, only device data compaction is performed if device data compaction is supported on the device. Otherwise, data compression is performed.

If *YES is specified for this parameter and *DEV is specified for the COMPACT parameter, both device data compaction and device data compression are performed if supported on the device.

***DEV** If the save is to tape and the target device supports compression, hardware compression is performed. Otherwise, no data compression is performed.

***NO** No data compression is performed.

***YES** If the save is to tape and the target device supports compression, hardware compression is performed. If compression is not supported, or if the save data is written to optical media or to a save file, software compression is performed. Low software compression is used for all devices except optical DVD, which uses medium software compression.

***LOW** If the save operation is to a save file or optical, software data compression is performed with the SNA algorithm. Low compression is usually faster and the compressed data is usually larger than if medium or high compression is used.

Note: This value is not valid for tape.

***MEDIUM**
If the save operation is to a save file or optical, software data compression is performed with the TERSE algorithm. Medium compression is usually slower than low compression but faster than high compression. The compressed data is usually smaller than if low compression is used and larger than if high compression is used.

Note: This value is not valid for tape.

***HIGH**
If the save operation is to a save file or optical, software data compression is performed with the LZ1 algorithm. High compression is usually slower and the compressed data is usually smaller than if low or medium compression is used.

Note: This value is not valid for tape.

Top

Data compaction (COMPACT)

Specifies whether device data compaction is performed.

***DEV** Device data compaction is performed if the data is saved to tape and all tape devices specified for the **Device (DEV)** parameter support the compaction feature.

Note: If *DEV is specified for both the **Data compression (DTACPR)** parameter and this parameter, only device data compaction is performed if device data compaction is supported on the device. Otherwise, data compression is performed if supported on the device.

If *YES is specified for the DTACPR parameter and *DEV is specified for this parameter, both device data compaction and device data compression are performed if supported on the device.

***NO** Device data compaction is not performed.

Top

Examples

Example 1: Tapes Cleared Automatically

```
SAVSYS   DEV(TAP01)  CLEAR(*ALL)
```

This command saves licensed internal code, system objects, all user profiles (including private authority for objects), and all line, controller, and device descriptions. They are saved on the tape put on the TAP01 tape drive. Each uncleared tape is automatically cleared when it is encountered, and the save operation continues without operator intervention.

Example 2: Operating Receives Message of Exceeded Storage Capacity

```
SAVSYS   DEV(TAP01)  VOL(ABCDE)
```

The system data is saved on the TAP01 tape drive, starting on the tape volume labeled ABCDE. If the save operation exceeds the storage capacity of one tape, a message requesting that another volume be put on the TAP01 tape drive is shown to the operator.

Example 3: Saving Data on Two Tape Drives in Alternating Order

```
SAVSYS   DEV(TAP01 TAP02)
```

The system data is saved on tape drives TAP01 and TAP02 in alternating order. If the save operation exceeds the storage capacity of two tapes, a message requesting that another volume be put on TAP01 is shown to the operator. The tapes are rewound at the completion of the save system operation.

Top

Error messages

***ESCAPE Messages**

CPF2206
User needs authority to do requested function on object.

CPF222E
&1 special authority is required.

CPF3703
&2 &1 in &3 not saved.

CPF3709
Tape devices do not support same densities.

CPF372B
Not all objects were saved.

CPF3727
Duplicate device &1 specified on device name list.

CPF3728
Device &1 specified with other devices.

CPF3733
&2 &1 in &3 previously damaged.

CPF3735
Storage limit exceeded for user profile &1.

CPF3738
Device &1 used for save or restore is damaged.

CPF3767
Device &1 not found.

CPF3768
Device &1 not valid for command.

CPF3772
SAVSYS completed. One or more objects not saved.

CPF3785
Not all subsystems ended.

CPF3793
Machine or ASP storage limit reached.

CPF3794
Save or restore operation ended unsuccessfully.

CPF3797
Objects from library &3 not saved. Save limit exceeded.

CPF3798
Installation &2 &1 in &3 not found.

CPF384E
USEOPTBLK(*YES) not valid for CD-ROM premastering.

CPF3873
Licensed program &1 option &2 release &4 not saved.

CPF388B
Optical file path name not valid.

CPF3894
Cancel reply received for message &1.

CPF38A4
ASP device &1 not correct.

CPF9814
Device &1 not found.

CPF9833
*CURASPGRP or *ASPGRPPRI specified and thread has no ASP group.

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.

CPF9850
Override of printer file &1 not allowed.

CPF9851
Overflow value for file &1 in &2 too small.

CPF9860
Error occurred during output file processing.

CPFB8ED
Device description &1 not correct for operation.

Top

Save System Information (SAVSYSINF)

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

Parameters
Examples
Error messages

The Save System Information (SAVSYSINF) command saves a subset of system data and objects saved by the Save System (SAVSYS) command. The system data and objects may be restored by the Restore System Information (RSTSYSINF) command.

The history information for the data area QSYSINF in QSYS is updated with the date, time, and place where the system information is saved.

SAVSYSINF is not to be considered a replacement for the SAVSYS command and is not to be used for system upgrades or migrations.

Objects saved from QSYS include:

- *JOBD
- *JOBQ
- *EDTD
- *JRN
- *SBSD
- *CLS
- *MSGQ
- *TBL
- *IGCTBL
- *DTAARA
- *CMD objects changed since the last SAVSYS
- *MSGF objects changed since the last SAVSYS

Additional items saved include:

- System reply list
- Service attributes
- Environment variables
- Certain system values
- Network attributes
- PTFs applied since the last SAVSYS

Some items NOT saved as part of SAVSYSINF command:

- Licensed Internal Code
- QSYS library
- Security objects (use the SAVSECDTA command)
- Configuration objects (use the SAVCFG command)
- User profiles (use the SAVSECDTA command)
- System values related to date/time
- System values which cannot be changed. For system values which can be changed, refer to the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/

- QPWDLVL (Password level) system value

Note: The SAVSYSINF command issues several save commands for saving objects. Parameters may or may not be used for all save commands.

Top

Parameters

Keyword	Description	Choices	Notes
DEV	Device	Single values: *SAVF Other values (up to 4 repetitions): *Name*	Required, Positional 1
VOL	Volume identifier	Single values: ***MOUNTED** Other values (up to 75 repetitions): *Character value*	Optional
SEQNBR	Sequence number	1-16777215, ***END**	Optional
EXPDATE	File expiration date	*Date*, ***PERM**	Optional
ENDOPT	End of media option	***REWIND**, *LEAVE, *UNLOAD	Optional
SAVF	Save file	*Qualified object name*	Optional
	Qualifier 1: Save file	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
OPTFILE	Optical file	*Path name*, *****	Optional
USEOPTBLK	Use optimum block	***YES**, *NO	Optional
CLEAR	Clear	***NONE**, *ALL, *AFTER, *REPLACE	Optional
DTACPR	Data compression	***DEV**, *NO, *YES, *LOW, *MEDIUM, *HIGH	Optional
COMPACT	Data compaction	***DEV**, *NO	Optional
OUTPUT	Output	***NONE**, *PRINT, *OUTFILE	Optional
OUTFILE	File to receive output	*Qualified object name*	Optional
	Qualifier 1: File to receive output	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
OUTMBR	Output member options	*Element list*	Optional
	Element 1: Member to receive output	*Name*, ***FIRST**	
	Element 2: Replace or add records	***REPLACE**, *ADD	

Top

Device (DEV)

Specifies the name of the device used for the save operation. The device name must already be known on the system by a device description.

This is a required parameter.

Single values

***SAVF** The save operation is done using the save file specified for the **Save file (SAVF)** parameter.

Other values

optical-device-name
Specify the name of the optical device used for the save operation.

tape-media-library-device-name
Specify the name of the tape media library device used for the save operation.

tape-device-name
Specify the names of one or more tape devices used for the save operation. If a virtual tape device is used, it must be the only device specified. If multiple tape devices are used, they must have compatible media formats and their names must be specified in the order in which they are used. Using more than one tape device permits one tape volume to be rewound and unloaded while another tape device processes the next tape volume.

Top

Volume identifier (VOL)

Specifies the volume identifiers of the volumes, or the cartridge identifiers of tapes in a tape media library device, on which the data is saved. The volumes must be placed in the device in the same order as specified for this parameter.

Single values

***MOUNTED**
The data is saved on the volumes placed in the device. For a media library device, the volume to be used is the next cartridge in the category mounted by the Set Tape Category (SETTAPCGY) command.

Note: This value cannot be specified when using an optical media library device.

Other values (up to 75 repetitions)

character-value
Specify the identifiers of one or more volumes in the order in which they are placed in a device and used to save the data.

Top

Sequence number (SEQNBR)

Specifies, when tape is used, the sequence number to use as the starting point for the save operation.

***END** The save operation begins after the last sequence number on the first tape. If the first tape is full, an error message is issued and the operation ends.

1-16777215
Specify the sequence number of the file to be used for the save operation.

Top

File expiration date (EXPDATE)

Specifies the expiration date of the file created by the save operation. If a date is specified, the file is protected and cannot be overwritten until the specified expiration date.

Notes:

1. This parameter is valid for tape and optical files.

2. Specifying this parameter does not protect against a later save operation specifying CLEAR(*ALL).

<u>*PERM</u>
The file is protected permanently.

date Specify the date when protection for the file ends.

Top

End of media option (ENDOPT)

Specifies the operation that is automatically done on the tape or optical volume after the save operation ends. If more than one volume is used, this parameter applies only to the last volume used; all other volumes are unloaded when the end of the volume is reached.

Note: This parameter is valid only if a tape or optical device name is specified for the DEV parameter. For optical devices, *UNLOAD is the only special value supported, *REWIND and *LEAVE will be ignored.

<u>*REWIND</u>
The tape is automatically rewound, but not unloaded, after the operation has ended.

***LEAVE**
The tape does not rewind or unload after the operation ends. It remains at the current position on the tape drive.

***UNLOAD**
The tape is automatically rewound and unloaded after the operation ends. Some optical devices will eject the volume after the operation ends.

Top

Save file (SAVF)

Specifies the save file that is used to contain the saved data. The save file must be empty, unless *ALL is specified for the **Clear (CLEAR)** parameter.

Note: A value must be specified for this parameter if *SAVF is specified for the **Device (DEV)** parameter.

Qualifier 1: Save file

name Specify the name of save file to be used.

Qualifier 2: Library

<u>*LIBL</u> 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 save file. If no current library entry exists in the library list, the QGPL library is used.

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

Top

Optical file (OPTFILE)

Specifies the path name of the optical file that is used for the save operation, beginning with the root directory of the volume.

For more information on specifying path names, refer to "Object naming rules" in the CL topic collection in the Programming category in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/.

*****
The system generates an optical file name in the root directory of the optical volume.

'optical-directory-path-name/*'
The system generates an optical file name in the specified directory of the optical volume.

Top

Use optimum block (USEOPTBLK)

Specifies whether or not the optimum block size is used for the save operation.

Note: Specifying USEOPTBLK(*YES) may result in a tape that can be duplicated only to a device that supports the same block size.

***YES**
The optimum block size supported by the device is used for Save commands. If the block size that is used is larger than a block size that is supported by all device types, then:

- Performance may improve.
- The tape file that is created is only compatible with a device that supports the block size used. Commands such as Duplicate Tape (DUPTAP) do not duplicate files unless the files are being duplicated to a device which supports the same block size that was used.
- The value for the DTACPR parameter is ignored.

***NO**
The optimum block size supported by the device is not used. Save commands use the default block size supported by all device types. The tape volume can be duplicated to any media format using the Duplicate Tape (DUPTAP) command.

Top

Clear (CLEAR)

Specifies whether active data on the media is automatically cleared or replaced. Active data is any file on the media that has not expired. For saves to tape, clearing active data will make any files on the tape volume beyond the last file written by the save operation no longer accessible. For saves to optical, the files written by the save operation can be automatically replaced while other files on the volume remain active, or all active files can be automatically cleared. Clearing does not erase the data, it just makes the files no longer accessible.

Notes:

1. Clearing a tape does not initialize it. You should initialize tapes to a standard label format before the save command is issued by using the Initialize Tape (INZTAP) command and specifying a value for the NEWVOL parameter.
2. Clearing an optical volume does initialize it.
3. If a volume that is not initialized is encountered during the save operation, an inquiry message is sent and an operator can initialize the volume.

***NONE**
None of the media is automatically cleared. If the save operation encounters active data on a tape volume or in a save file, an inquiry message is sent, allowing the operator to either end the save operation or clear the media. If the save operation encounters the specified optical file, an inquiry message is sent, allowing the operator to either end the save operation or replace the file.

***ALL**
All of the media is automatically cleared.

If tapes are used and a sequence number is specified for the SEQNBR parameter, the first tape is cleared beginning at that sequence number. All tapes following that first tape are completely cleared. To clear the entire first tape, SEQNBR(1) must be specified.

***AFTER**
All media after the first volume is automatically cleared. If the save operation encounters active data on the first tape volume, an inquiry message is sent, allowing the operator to either end the save operation or clear the media. If the save operation encounters the specified optical file on the first volume, an inquiry message is sent, allowing the operator to either end the save operation or replace the file.

Note: The *AFTER value is not valid for save files.

***REPLACE**
Active data on the media is automatically replaced. Optical volumes are not initialized. Other media is automatically cleared in the same way as the *ALL value.

Top

Data compression (DTACPR)

Specifies whether data compression is used. If the save is running while other jobs on the system are active and software compression is used, the overall system performance may be affected.

Note: If *DEV is specified for both this parameter and the **Data compaction (COMPACT)** parameter, only device data compaction is performed if device data compaction is supported on the device. Otherwise, data compression is performed.

If *YES is specified for this parameter and *DEV is specified for the COMPACT parameter, both device data compaction and device data compression are performed if supported on the device.

***DEV** If the save is to tape and the target device supports compression, hardware compression is performed. Otherwise, no data compression is performed.

***NO** No data compression is performed.

***YES** If the save is to tape and the target device supports compression, hardware compression is performed. If compression is not supported, or if the save data is written to optical media or to a save file, software compression is performed. Low software compression is used for all devices except optical DVD, which uses medium software compression.

***LOW** If the save operation is to a save file or optical, software data compression is performed with the SNA algorithm. Low compression is usually faster and the compressed data is usually larger than if medium or high compression is used.

Note: This value is not valid for tape.

***MEDIUM**
If the save operation is to a save file or optical, software data compression is performed with the TERSE algorithm. Medium compression is usually slower than low compression but faster than high compression. The compressed data is usually smaller than if low compression is used and larger than if high compression is used.

Note: This value is not valid for tape.

***HIGH**
If the save operation is to a save file or optical, software data compression is performed with the LZ1 algorithm. High compression is usually slower and the compressed data is usually smaller than if low or medium compression is used.

Note: This value is not valid for tape.

Top

Data compaction (COMPACT)

Specifies whether device data compaction is performed.

***DEV** Device data compaction is performed if the data is saved to tape and all tape devices specified for the **Device (DEV)** parameter support the compaction feature.

Note: If *DEV is specified for both the **Data compression (DTACPR)** parameter and this parameter, only device data compaction is performed if device data compaction is supported on the device. Otherwise, data compression is performed if supported on the device.

If *YES is specified for the DTACPR parameter and *DEV is specified for this parameter, both device data compaction and device data compression are performed if supported on the device.

***NO** Device data compaction is not performed.

Top

Output (OUTPUT)

Specifies whether a list with information about the saved objects is created. The information can be printed with the job's spooled output or directed to a database file.

***NONE**
No output listing is created.

***PRINT**
The output is printed with the job's spooled output.

***OUTFILE**
The output is directed to the database file specified for the **File to receive output (OUTFILE)** parameter.

Note: You must specify a database file name for the **File to receive output (OUTFILE)** parameter when OUTPUT(*OUTFILE) is specified.

Top

File to receive output (OUTFILE)

Specifies the database file to which the output of the command is directed. If the file does not exist, this command creates a database file in the specified library. If the file is created, the public authority for the file is the same as the create authority specified for the library in which the file is created. Use the Display Library Description (DSPLIBD) command to show the library's create authority.

Qualifier 1: File to receive output

name Specify the name of the database file to which the command output is directed.

Qualifier 2: Library

***LIBL** The library list is used to locate the file. If the file is not found, one is created in the current library. If no current library exists, the file will be created in the QGPL library.

***CURLIB**
The current library for the thread is used to locate the 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.

Note: If a new file is created, the system uses the IBM-supplied file QASAVOBJ with format name QSRSAV as a model.

Top

Output member options (OUTMBR)

Specifies the name of the database file member to which the output is directed when *OUTFILE is specified for the **Output (OUTPUT)** parameter.

Element 1: Member to receive output

***FIRST**
The first member in the file receives the output. If OUTMBR(*FIRST) is specified and the member does not exist, the system creates a member with the name of the file specified for the **File to receive output (OUTFILE)** parameter.

name Specify the name of the file member that receives the output. If OUTMBR(member-name) is specified and the member does not exist, the system creates it.

If the member exists, you can add records to the end of the existing member or clear the existing member and add the records.

Element 2: Replace or add records

***REPLACE**
The existing records in the specified database file member are replaced by the new records.

***ADD** The new records are added to the existing information in the specified database file member.

Top

Examples

Example 1: Tapes Cleared Automatically

```
SAVSYSINF   DEV(TAP01)  CLEAR(*ALL)
```

This command saves the system information on the tape put on the TAP01 tape drive. Each uncleared tape is automatically cleared when it is encountered, and the save operation continues without operator intervention.

Example 2: Save File Cleared Automatically and Output Generated to a File

```
SAVSYSINF   DEV(*SAVF)  SAVF(QGPL/SAVF)  CLEAR(*ALL)
            OUTPUT(*OUTFILE)  OUTFILE(QGPL/OUTPUT)
```

This command saves the system information to the save file named SAVF in library QGPL. The save file will be cleared automatically. Information about what was saved will be written to the first member of the file named OUTPUT in library QGPL; the file and member will be created if they do not exist.

Top

Error messages

***ESCAPE Messages**

CPF38A7
SAVSYSINF completed. One or more objects not saved.

CPD37AD
Save file not found for PTF &1-&2 &3.

Top

Submit Data Base Jobs (SBMDBJOB)

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

Parameters
Examples
Error messages

The Submit Database Jobs (SBMDBJOB) command allows you to submit jobs to job queues so they can be run as batch jobs. The input stream is read either from a physical database file or from a logical database file that has a single-record format. This command allows you to specify the name of this database file and its member, the name of the job queue to be used, and to decide whether jobs being submitted can be displayed by the Work with Submitted Jobs (WRKSBMJOB) command

A Submit Database Jobs operation reads the file once and ends when the end-of-file is read or when an End Input (ENDINP) command is encountered. The ENDINP command (a delimiter) is not recognized if it is within an inline file that ends with characters that are not default ending characters (as specified in the ENDCHAR parameter of the Data (DATA) command). The SBMDBJOB operation can be canceled either by canceling the request from the system request menu or by canceling the job in which the process is running.

In contrast to a spool reader started with the Start Database Reader (STRDBRDR) command, the SBMDBJOB command operates in the same process as the requesting function and does not do syntax checking on the input stream.

Restriction: The specified database file either must consist of single-field records and must have an arrival sequence access path, or it must be a standard database source file.

Top

Parameters

Keyword	Description	Choices	Notes
FILE	Data base file	*Qualified object name*	Required, Positional 1
	Qualifier 1: Data base file	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
MBR	Member	*Name*, ***FIRST**	Optional, Positional 2
JOBQ	Job queue	*Qualified object name*	Optional, Positional 3
	Qualifier 1: Job queue	*Name*, **QBATCH**	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
DSPSBMJOB	Show on submitted job list	***YES**, *NO	Optional

Top

Data base file (FILE)

Specifies the database file from which the input stream is read.

This is a required parameter.

Qualifier 1: Data base file

name Specify the name of the database file that contains the input stream that you want to read.

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 name of the library where the file is located.

Top

Member (MBR)

Specifies the member in the specified file that contains the input stream to be read.

***FIRST**
No member name is specified; the first member in the file is used.

name Specify the name of the member that contains the input stream to read.

Top

Job queue (JOBQ)

Specifies the job queue on which the job entries are placed. A job entry is placed on this queue for each job in the input stream that has JOBQ(*RDR) specified on the Batch Job (BCHJOB) command. If *RDR is not specified on the BCHJOB command, the job queue specified on the BCHJOB command or in the job description is used. (The job queue for each job in the input stream can be different.) This parameter is valid only if ACTION(*SUBMIT) is specified in the existing network job entry or in a subsequent Change Network Job Entry (CHGNETJOBE) command.

Note: If both the user identified in the job description of the job being read and the user processing the Submit Database Job (SBMDBJOB) command are not authorized to the job queue on which the job should be placed, the job ends and a diagnostic message is placed in the job log. The input stream, continues to be processed, starting with the next job. If either user is authorized to the job queue, the job runs without error.

Qualifier 1: Job queue

QBATCH
The job entry is to be placed on the QBATCH job queue, which is the default job queue. This will happen if *RDR is specified on the **Job queue (JOBQ)** parameter of the Batch Job (BCHJOB) command. The Batch Job (BCHJOB) command is contained in the input stream itself.

name Specify the name of the job queue to which each job in the input stream will be sent. This will happen if *RDR is specified on the JOBQ parameter of the Batch Job (BCHJOB) 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 job queue.

name Specify the name of the library where the job queue is located.

Top

Show on submitted job list (DSPSBMJOB)

Specifies whether the jobs being submitted are displayed on the submitted jobs display. If *YES is specified here, these submitted jobs can be displayed when the Work with Submitted Jobs (WRKSBMJOB) command is used.

***YES** This job can be displayed by the Work with Submitted Jobs (WRKSBMJOB) command.

***NO** This job is not displayed on any display produced by the Work with Submitted Jobs (WRKSBMJOB) command.

Top

Examples

```
SBMDBJOB   FILE(QGPL/BILLING)
```

This command submits jobs using input from the database file named BILLING, which is in the QGPL library. The first member in the BILLING file contains the input stream to be processed. The default system-supplied job queue QBATCH is used.

Top

Error messages

***ESCAPE Messages**

CPF1751
Error while processing job &3/&2/&1.

CPF1754
File &1 in library &2 not database file or DDM file.

CPF1760
Submit jobs command not allowed.

CPF2207
Not authorized to use object &1 in library &3 type *&2.

CPF3307
Job queue &1 in &2 not found.

CPF3330
Necessary resource not available.

CPF3363
Message queue &1 in library &2 not found.

CPF9802
Not authorized to object &2 in &3.

CPF9812
File &1 in library &2 not found.

CPF9815
Member &5 file &2 in library &3 not found.

CPF9845
Error occurred while opening file &1.

CPF9846
Error while processing file &1 in library &2.

***STATUS Messages**

CPF1762
Reading job &3/&2/&1.

Top

Submit Finance Job (SBMFNCJOB)

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

Parameters
Examples
Error messages

The Submit Finance Job (SBMFNCJOB) command submits a batch job that lets your finance application programs communicate with your 4701 or 4702 controller application programs.

Use the SBMFNCJOB command only if:

- Communicating with a 4701 or 4702 control unit
- A device table and a program table have been defined using the Work with Device Table (WRKDEVTBL) and Work with Program Table (WRKPGMTBL) commands; defining a user table using the Work with User Table (WRKUSRTBL) command is optional
- The user's 4701 or 4702 control unit application program sends data (transactions) first and expects a response
- The user's 4701 or 4702 control unit application program passes data in the proper format

Restriction: This command is shipped with public *EXCLUDE authority.

Top

Parameters

Keyword	Description	Choices	Notes
DEVTBL	Device table	*Name*	Required, Positional 1
PGMTBL	Program table	*Name*	Required, Positional 2
USRTBL	User table	*Name*, ***NONE**	Optional, Positional 3
JOB	Job name	*Name*, **QFNCJOB**	Optional
JOBD	Job description	*Qualified object name*	Optional
	Qualifier 1: Job description	*Name*, **QFNC**	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
MSGQ	Message queue	Single values: ***WRKSTN**, *NONE Other values: *Qualified object name*	Optional
	Qualifier 1: Message queue	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	

Top

Device table (DEVTBL)

Specifies the name of the device table that the finance job uses to determine which 4704 or 3624 devices it controls. This is a required parameter.

Top

Program table (PGMTBL)

Specifies the name of the program table that the finance job uses to determine, from the program ID (sent in the data stream with a finance transaction), which user program names will process the finance transaction. This is a required parameter.

Top

User table (USRTBL)

Specifies the name of the user table that the finance job uses to verify a valid user when a finance sign-on is received.

The possible values are:

***NONE**
No user IDs are verified.

user-table-name
Specify the name of a user table that defines user IDs for the 4700 device.

Top

Job name (JOB)

Specifies the job name that is associated with the submitted finance job.

The possible values are:

QFNCJOB
The job name is submitted as QFNCJOB.

job-name
Specify the user-defined job name that is associated with the submitted finance job.

Top

Job description (JOBD)

Specifies the name and library of the job description that is used by the finance job.

The possible values are:

***QFNC**
The submitted finance job uses the job description QFNC.

job-description-name
Specify the name and library of a job description that is used by the finance job. (If no library name is given, the job description is found through the library list used by the job in which the Submit Finance Job (SBMFNCJOB) command is entered.

The possible library values are:

***LIBL**

The library list is used to locate the finance job.

***CURLIB**

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

library-name

Specify the name of the library where the finance job is located.

Top

Message queue (MSGQ)

Specifies the name of the message queue, to which messages can be sent while the finance job is running.

The possible values are:

WRKSTN
The finance messages are sent to the message queue of the work station from which the finance job was submitted.

***NONE**
No finance messages are sent to a message queue.

message-queue-name
Specify the name and library of the user-defined message queue to which messages are sent. (If no library name is given, the library list of the job issuing the Submit Finance Job (SBMFNCJOB) command is used to find the queue.)

The possible library values are:

***LIBL** The library list is used to find the name of the message queue.

***CURLIB**
The current library for the job is used to find the name of the message queue. 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 name of the message queue is located.

Top

Examples

Example 1: Submitting a Batch Job that Communicates with Devices it Acquires

```
SBMFNCJOB   DEVTBL(DEVTBL1)  PGMTBL(PGMTBL1)  USRTBL(USRTBL1)
```

This command submits batch job QFNCJOB. The job communicates with all devices it acquires from device table DEVTBL1, allowing users whose user IDs are found in USRTBL1 to sign on the devices. Each transaction sent by the finance devices is processed by determining, in PGMTBL1, which application program must be called, then it calls that program.

The job description used by the finance job in this example is QFNC. Messages sent as a result of the finance job are sent to the message queue of the work station from which the job was submitted.

Example 2: User IDs Not Verified

```
SBMFNCJOB   DEVTBL(DEVTBL2)  PGMTBL(PGMTBL2)
            JOB(CTFJOB)  JOBD(CTFJOBD)  MSGQ(*NONE)
```

This command submits batch job CTFJOB. CTFJOB runs under job description CTFJOBD and does not send messages to any work station message queue while running. No verification of user IDs is performed by the finance job.

Top

Error messages

***ESCAPE Messages**

CPF8382

Finance job cannot be processed.

Top

Submit Job (SBMJOB)

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

Parameters
Examples
Error messages

The Submit Job (SBMJOB) command allows a job that is running to submit another job to a job queue to be run later as a batch job. Only one element of request data can be placed on the new job's message queue. The request data can be a CL command if the routing entry used for the job specifies a CL command processing program (such as the IBM-supplied QCMD program).

Restrictions:

1. The user that issues the SBMJOB command must have:
   - use (*USE) authority to the user profile specified on the **User (USER)** parameter.
   - use (*USE) authority to the command specified on the **Command to run (CMD)** parameter and execute (*EXECUTE) authority to the library containing that command.
   - use (*USE) authority to the job description specified on the **Job description (JOBD)** parameter and execute (*EXECUTE) authority to the library containing that job description.
   - use (*USE) authority to the job queue specified on the **Job queue (JOBQ)** parameter and execute (*EXECUTE) authority to the library containing that job queue.
   - use (*USE) and add (*ADD) authority to the message queue specified on the **Message queue (MSGQ)** parameter and execute (*EXECUTE) authority to the library containing that message queue.
   - job control (*JOBCTL) special authority to use the **Submitted for (SBMFOR)** parameter.
   - use (*USE) authority to the Change Accounting Code (CHGACGCDE) command in order to specify a character-value accounting code on the **Accounting code (ACGCDE)** parameter.
2. The user for the submitted job must have:
   - use (*USE) authority to the job description specified on the **Job description (JOBD)** parameter.
   - read (*READ) authority to the output queue specified on the **Output queue (OUTQ)** parameter and execute (*EXECUTE) authority to the library, in the submitted job's name space, that contains that output queue.
   - use (*USE) authority to all the libraries specified on the **System library list (SYSLIBL)** parameter.
   - use (*USE) authority to the library specified on the **Current library (CURLIB)** parameter.
   - use (*USE) authority to all the libraries specified on the **Initial library list (INLLIBL)** parameter.
   - use (*USE) authority to all auxiliary storage pool (ASP) device descriptions specified on the **Initial ASP group (INLASPGRP)** parameter.
   - use (*USE) authority to the sort sequence table specified on the **Sort sequence (SRTSEQ)** parameter and execute (*EXECUTE) authority to the library, in the submitted job's name space, that contains that sort sequence table.
3. This command is conditionally threadsafe. If a Job Notification Exit Point has been registered to send a message to a DDM data queue whenever a Submit Job is done, the message will not be sent if SBMJOB command is issued in a multithreaded job. For more information on the Job Notification function, refer to the Job Notification Exit Point in the Work Management chapter of the System API Reference manual.

Top

Parameters

Keyword	Description	Choices	Notes
CMD	Command to run	*Command string*	Optional
JOB	Job name	*Name*, ***JOBD**	Optional, Positional 1
JOBD	Job description	Single values: ***USRPRF** Other values: *Qualified object name*	Optional, Positional 2
	Qualifier 1: Job description	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
JOBQ	Job queue	Single values: ***JOBD** Other values: *Qualified object name*	Optional, Positional 4
	Qualifier 1: Job queue	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
JOBPTY	Job priority (on JOBQ)	1-9, ***JOBD**	Optional
OUTPTY	Output priority (on OUTQ)	1-9, ***JOBD**	Optional
PRTDEV	Print device	*Name*, ***CURRENT**, *USRPRF, *SYSVAL, *JOBD	Optional
OUTQ	Output queue	Single values: ***CURRENT**, *USRPRF, *DEV, *JOBD Other values: *Qualified object name*	Optional
	Qualifier 1: Output queue	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
USER	User	*Name*, ***CURRENT**, *JOBD	Optional, Positional 3
ACGCDE	Accounting code	*Character value*, ***CURRENT**, *USRPRF, *JOBD	Optional
PRTTXT	Print text	*Character value*, ***CURRENT**, *JOBD, *SYSVAL, *BLANK	Optional
RTGDTA	Routing data	*Character value*, **QCMDB**, *JOBD, *RQSDTA	Optional
RQSDTA	Request data or command	*Character value*, ***CMD**, *JOBD, *NONE, *RTGDTA	Optional
SYSLIBL	System library list	***CURRENT**, *SYSVAL	Optional
CURLIB	Current library	*Name*, ***CURRENT**, *USRPRF, *CRTDFT	Optional
INLLIBL	Initial library list	Single values: ***CURRENT**, *JOBD, *SYSVAL, *NONE Other values (up to 250 repetitions): *Name*	Optional
INLASPGRP	Initial ASP group	*Name*, ***CURRENT**, *JOBD, *NONE	Optional
LOG	Message logging	*Element list*	Optional
	Element 1: Level	0-4, ***JOBD**	
	Element 2: Severity	0-99, ***JOBD**	
	Element 3: Text	***JOBD**, *MSG, *SECLVL, *NOLIST	
LOGCLPGM	Log CL program commands	***JOBD**, *NO, *YES	Optional
LOGOUTPUT	Job log output	***JOBD**, *SYSVAL, *JOBLOGSVR, *JOBEND, *PND	Optional
JOBMSGQMX	Job message queue maximum size	2-64, ***JOBD**, *SYSVAL	Optional
JOBMSGQFL	Job message queue full action	***JOBD**, *SYSVAL, *NOWRAP, *WRAP, *PRTWRAP	Optional
INQMSGRPY	Inquiry message reply	***JOBD**, *RQD, *DFT, *SYSRPYL	Optional
HOLD	Hold on job queue	***JOBD**, *NO, *YES	Optional
SCDDATE	Schedule date	*Date*, ***CURRENT**, *MONTHSTR, *MONTHEND, *MON, *TUE, *WED, *THU, *FRI, *SAT, *SUN	Optional
SCDTIME	Schedule time	*Time*, ***CURRENT**	Optional
DATE	Job date	*Date*, ***JOBD**, *SYSVAL	Optional
SWS	Job switches	*Character value*, ***JOBD**	Optional

Keyword	Description	Choices	Notes
DSPSBMJOB	Allow display by WRKSBMJOB	***YES**, *NO	Optional
SBMFOR	Submitted for	Single values: ***CURRENT** Other values: *Qualified job name*	Optional
	Qualifier 1: Submitted for	*Name*	
	Qualifier 2: User	*Name*	
	Qualifier 3: Number	000000-999999	
MSGQ	Message queue	Single values: ***USRPRF**, *WRKSTN, *NONE Other values: *Qualified object name*	Optional
	Qualifier 1: Message queue	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
SRTSEQ	Sort sequence	Single values: ***CURRENT**, *USRPRF, *SYSVAL, *HEX, *LANGIDUNQ, *LANGIDSHR Other values: *Qualified object name*	Optional
	Qualifier 1: Sort sequence	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
LANGID	Language ID	*Character value*, ***CURRENT**, *USRPRF, *SYSVAL	Optional
CNTRYID	Country or region ID	*Character value*, ***CURRENT**, *USRPRF, *SYSVAL	Optional
CCSID	Coded character set ID	1-65535, ***CURRENT**, *USRPRF, *SYSVAL, *HEX	Optional
CPYENVVAR	Copy environment variables	***NO**, *YES	Optional
ALWMLTTHD	Allow multiple threads	***JOBD**, *NO, *YES	Optional
SPLFACN	Spooled file action	***CURRENT**, *JOBD, *SYSVAL, *KEEP, *DETACH	Optional

Top

Command to run (CMD)

Specifies a command that runs in the batch job, if the routing program used when this batch job is initiated is the IBM-supplied default routing program (QCMD). Because this command is used for the request data, this parameter is mutually exclusive with the **Request data or command (RQSDTA)** parameter.

command-string
Specify the command that is run in the batch job. The command cannot be longer than 20000 characters.

Top

Job name (JOB)

Specifies the name that is associated with the job when it is processed by the system.

***JOBD**
The simple name of the job description used with this job is the name of the job itself.

name Specify the simple name of the job that is used while it is being processed by the system.

Top

Job description (JOBD)

Specifies the job description used with the job.

Single values

***USRPRF**
The job description in the user profile under which the submitted job initially runs is used as the job description of the submitted job.

Qualifier 1: Job description

name Specify the name of the job description used for the job.

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 job description. If no library is specified as the current library for the thread, the QGPL library is used.

name Specify the library where the job description is located.

Top

Job queue (JOBQ)

Specifies the job queue in which this job is placed.

Single values

***JOBD**
The submitted job is placed on the job queue named in the specified job description.

Qualifier 1: Job queue

name Specify the name of the job queue.

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 job queue. If no library is specified as the current library for the thread, the QGPL library is used.

name Specify the library where the job queue is located.

Top

Job priority (on JOBQ) (JOBPTY)

Specifies the job queue scheduling priority. The highest priority is 1 and the lowest priority is 9.

***JOBD**
The scheduling priority specified in the job description is used for the job.

1-9 Specify the scheduling priority for the job.

Top

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.

***JOBD**
The output priority specified in the job description is used for the job.

1-9 Specify the priority of the submitted job's output files.

Top

Print device (PRTDEV)

Specifies the qualified name of the default printer device for this job. If the printer file being used to create the output specifies to spool the file, 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 on the **Output queue (OUTQ)** parameter of the printer file, job description, user profile and work station.

***CURRENT**
The same printer device being used by the job that is currently running is used by the submitted job.

***USRPRF**
The printer device specified in the user profile under which the submitted job initially runs is used as the printer device for this job. The printer device name is obtained from the profile when this command is run.

***SYSVAL**
The printer device specified in the system value, QPRTDEV, when this command is run is used for the submitted job.

***JOBD**
The printer device specified in the job description is used for the submitted job.

name Specify the name of the printer device used for the submitted job.

Top

Output queue (OUTQ)

Specifies the qualified name of the output queue used for spooled files that specify OUTQ(*JOB). This parameter applies only to printer files that have *JOB specified on the OUTQ parameter.

Single values

***CURRENT**
The output queue used by the job that is currently running is used for the submitted job.

***USRPRF**
The output queue in the user profile under which the submitted job initially runs is used as the output queue for the submitted job. The output queue name is obtained from the profile when this command is run.

***DEV** The output queue associated with the printer specified on the **Device (DEV)** parameter of the printer file is used. The output queue has the same name as the printer. The printer file DEV parameter is determined by the Create Printer File (CRTPRTF), Change Printer File (CHGPRTF), or the Override with Printer File (OVRPRTF) commands.

Note: This assumes the defaults were specified on the OUTQ parameter for the printer file, job description, user profile, and workstation.

***JOBD**
The output queue named in the job description used with the submitted job is the job's default output queue.

Qualifier 1: Output queue

name Specify the name of the output queue that is used as the default output queue by the submitted job.

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 output queue. If no library is specified as the current library of the thread, the QGPL library is used.

name Specify the library where the output queue is located.

Top

User (USER)

Specifies the name of the user profile for the job being submitted. If *RQD is specified in the job description, *JOBD cannot be specified. Instead, *CURRENT or a user-name must be specified.

Note: The following IBM-supplied objects are not valid for this parameter:

- QDBSHR
- QDFTOWN
- QDOC
- QLPAUTO
- QLPINSTALL
- QRJE
- QSECOFR
- QSPL
- QSYS
- QTSTRQS

***CURRENT**
The same user profile used by the job that is currently running is used for the submitted job.

***JOBD**
The user profile named in the specified job description is used for the job being submitted.

name Specify the name of the user profile that is used for the job being submitted.

Top

Accounting code (ACGCDE)

Specifies the accounting code that is used by the submitted job when logging the usage of system resources in the system accounting journal QACGJRN.

***CURRENT**
The accounting code used by the job that is currently running is used for the submitted job.

***USRPRF**
The accounting code specified in the user profile under which the submitted job initially runs is used.

***JOBD**
The accounting code specified in the job description associated with the submitted job is used.

character-value
Specify the 15-character accounting code to be used by the submitted job. If less than 15 characters are specified, the string is padded on the right with blanks.

Top

Print text (PRTTXT)

Specifies the text that is printed at the bottom of each page of printed output and on separator pages.

***CURRENT**
The same print text of the submitting job is used.

***JOBD**
The value in the job description is used for this job.

***SYSVAL**
The system value, QPRTTXT, is used.

***BLANK**
No text is printed.

character-value
Specify the character string that is printed at the bottom of each page. A maximum of 30 characters can be entered, enclosed in apostrophes.

Top

Routing data (RTGDTA)

Specifies the routing data used to start the next routing step in the job. The routing data is used to determine the routing entry that identifies the program that the job runs.

QCMDB
The routing data used by the IBM-supplied subsystems to route batch jobs to the IBM-supplied control language processor QCMD is used.

***JOBD**
The routing data specified in the job description is used.

***RQSDTA**
The first 80 characters of the request data specified in the **Request data or command (RQSDTA)** parameter of this command are used as the routing data for the job.

character-value
Specify the character string that is used as the routing data for starting the job. A maximum of 80 characters can be entered, enclosed in apostrophes if necessary.

Top

Request data or command (RQSDTA)

Specifies the request data that is placed as the last entry in this job's message queue. The request data can be a CL command to be run or a string of characters used by another program.

***CMD** The input from the **Command to run (CMD)** parameter is placed in this job's message queue.

***JOBD**
The request data specified in the job description used by the job is placed in this job's message queue.

***NONE**
No request data is placed in the job's message queue.

***RTGDTA**
The routing data specified in the **Routing data (RTGDTA)** parameter of this command is placed as the last entry in the job's message queue.

character-value
Specify the character string that is placed as the last entry in the submitted job's message queue. A maximum of 3000 characters can be entered, enclosed in apostrophes if necessary.

Top

System library list (SYSLIBL)

Specifies the system portion of the initial library list that is used by the submitted job.

***CURRENT**
The same system library list being used by the thread that is currently running is used for the submitted job.

***SYSVAL**
The system default system library list is used by the job. It contains the library names that were specified in the system value, QSYSLIBL, at the time that the job is started.

Top

Current library (CURLIB)

Specifies the name of the current library associated with the submitted job.

***CURRENT**
The current library being used by the thread that is currently running is used for the submitted job.

***USRPRF**
The current library in the user profile under which the submitted job initially runs is used as the current library for the submitted job.

***CRTDFT**
There is no current library for the submitted job. If objects are created in the current library, QGPL is used as the default current library.

name Specify the library used as the current library of the submitted job.

Top

Initial library list (INLLIBL)

Specifies the initial user part of the library list that is used to search for any object names that were specified without a library qualifier.

Note: Duplication of library names in the library list is not allowed.

Single values

***CURRENT**
The library list being used by the thread that is currently running is used for the submitted job.

***JOBD**
The library list in the job description used with this job is used as the initial user part of the library list for the job.

***SYSVAL**
The system default user library list is used by this job. It contains the library names that were specified in the system value QUSRLIBL at the time that the job is started.

***NONE**
The user portion of the initial library list is empty.

Other values (up to 250 repetitions)

name Specify the names of one or more libraries that are the user portion of the library list and are used by this job. The libraries are searched in the same order as they are listed.

Top

Initial ASP group (INLASPGRP)

Specifies the initial setting for the auxiliary storage pool (ASP) group name for the initial thread of the submitted job. A thread can use the Set ASP Group (SETASPGRP) command to change its library name space. When an ASP group is associated with a thread, all libraries in the independent ASPs in the ASP group are accessible and objects in those libraries can be referenced using regular library-qualified object name syntax. The libraries in the independent ASPs in the specified ASP group plus the libraries in the system ASP (ASP number 1) and basic user ASPs (ASP numbers 2-32) form the library name space for the thread.

***CURRENT**
The ASP group name for the current thread is used for the submitted job.

***JOBD**
The initial ASP group name specified in the job description is used for the submitted job.

***NONE**
Specifies the initial thread of the submitted job will be started with no ASP group. The library name space will not include libraries from any ASP group. Only the libraries in the system ASP and any basic user ASPs will be in the library name space.

name Specify the name of the ASP group to be set for the initial thread of the submitted job. The ASP group name is the name of the primary ASP device within the ASP group. All libraries from all ASPs in this ASP group will be included in the library name space.

Top

Message logging (LOG)

Specifies the message logging values used to determine the amount and type of information sent to the job log by this job. This parameter has three elements: the message (or logging) level, the message severity, and the level of message text.

Element 1: Level

***JOBD**
The value specified for message logging in the job description is used for this job.

0-4 Specify the message logging level used for this job's messages. The possible logging levels are:

0 No messages are logged.

1 All messages sent to the job's external message queue with a severity greater than or equal to the message logging severity are logged. This includes the indications of job start, job end, and job completion status.

2 The following information is logged:

- Logging level 1 information
- Request messages which result in a high-level message with a severity code greater than or equal to the message logging severity. Both the request message and all associated messages are logged.

Note: A high-level message is one that is sent to the program message queue of the program that receives the request message. For example, QCMD is an IBM-supplied request processing program that receives request messages.

3 The following information is logged:

- Logging level 1 and 2 information
- All request messages
- Commands run by a CL program are logged if it is allowed by the logging of CL programs job attribute and the log attribute of the CL program.

4 The following information is logged:

- All request messages and all messages with a severity greater than or equal to the message logging severity, including trace messages.
- Commands run by a CL program are logged if it is allowed by the logging of CL programs job attribute and the log attribute of the CL program.

Element 2: Severity

***JOBD**
The value specified for message logging in the job description is used for this job.

0-99 Specify the message severity that is used in conjunction with the logging level to determine which error messages are logged in the job log.

Element 3: Text

***JOBD**
The value specified for message logging in the job description is used for this job.

***MSG** Only the message text is written to the job log.

***SECLVL**
Both the message text and the message help (cause and recovery) of the error message are written to the job log.

***NOLIST**
If the job ends normally, no job log is produced. If the job ends abnormally (if the job end code is 20 or higher), a job log is produced. The messages that appear in the job log contain both the message text and the message help.

Top

Log CL program commands (LOGCLPGM)

Specifies whether the commands that are run in a control language program are logged to the job log by way of the CL program's message queue. This parameter sets the status of the job's logging flag. If *JOB has been specified for the **Message logging (LOG)** parameter in the Create CL Program (CRTCLPGM) command, the flag set in the **Log CL program commands (LOGCLPGM)** parameter is used. Other values for the **Message logging (LOG)** parameter override the **Log CL program commands (LOGCLPGM)** parameter. The commands are logged in the same manner as the requests.

***JOBD**
The value specified in the job description is used.

***NO** The commands in a CL program are not logged to the job log.

***YES** The commands in a CL program are logged to the job log.

Top

Job log output (LOGOUTPUT)

Specifies how the job log will be produced when the job completes. This does not affect job logs produced when the message queue is full and the job message queue full action specifies *PRTWRAP. Messages in the job message queue are written to a spooled file, from which the job log can be printed, unless the Control Job Log Output (QMHCTLJL) API was used in the job to specify that the messages in the job log are to be written to a database file.

The job log output value can be changed at any time until the job log has been produced or removed. To change the job log output value for a job, use the Change Job (QWTCHGJB) API or the Change Job (CHGJOB) command.

The job log can be displayed at any time until the job log has been produced or removed. To display the job log, use the Display Job Log (DSPJOBLOG) command.

The job log can be removed when the job has completed and the job log has not yet been produced or removed. To remove the job log, use the Remove Pending Job Log (QWTRMVJL) API or the End Job (ENDJOB) command.

***JOBD**
The value specified in the job description is used.

***SYSVAL**
The value specified in the system value QLOGOUTPUT is used.

***JOBLOGSVR**
The job log will be produced by a job log server. For more information about job log servers, refer to the Start Job Log Server (STRLOGSVR) command.

***JOBEND**
The job log will be produced by the job itself. If the job cannot produce its own job log, the job log will be produced by a job log server. For example, a job does not produce its own job log when the system is processing a Power Down System (PWRDWNSYS) command.

***PND** The job log will not be produced. The job log remains pending until removed.

Top

Job message queue maximum size (JOBMSGQMX)

Specifies the maximum size of the job message queue.

***JOBD**
The value specified in the job description determines maximum size of the job message queue.

***SYSVAL**
The value in system value QJOBMSGQMX at the time the job is started is used as the maximum size of the job message queue.

2-64 Specify the maximum size, in megabytes, of the job message queue.

Top

Job message queue full action (JOBMSGQFL)

Specifies the action that should be taken when the job message queue is full.

***JOBD**
The value specified in the job description determines the action that should be taken.

***SYSVAL**
The value specified for the QJOBMSGQFL system value is used.

***NOWRAP**
The message queue does not wrap when it is full. This action ends the job.

***WRAP**
The message queue wraps to the start of the message queue when full and starts filling the message queue again.

***PRTWRAP**
The message queue wraps the job message queue when full and prints the messages that are being overlaid because of wrapping.

Top

Inquiry message reply (INQMSGRPY)

Specifies the way that predefined messages that are sent as a result of running this job are answered. You can specify that the inquiry message reply control is taken from the job description, or that all inquiry messages require a reply, or that a default reply is issued, or that the system reply list is checked for a matching reply as each predefined inquiry message is sent.

***JOBD**
The inquiry message reply control specified in the job description used with this job is used.

***RQD** A reply is required by the receiver of the inquiry message for all inquiry messages that occur when this command is run.

***DFT** The default message reply is used to answer any inquiry messages issued when this command is run.

***SYSRPYL**
The system reply list is checked to see if there is an entry for any inquiry message that is issued as a result of running this job that has a message identifier and any comparison data that match the inquiry message identifier and message data. If a match occurs, the reply value in that entry is used. If no entry exists for that message, a reply is required.

Top

Hold on job queue (HOLD)

Specifies whether this job is held at the time that it is put on the job queue. A job placed on the job queue in the hold state is held until it is released by the Release Job (RLSJOB) command or ended, either by the End Job (ENDJOB) command or by the Clear Job Queue (CLRJOBQ) command.

***JOBD**
The value specified in the job description determines whether the job is held when it is put on the job queue.

***NO** The job is not held when it is put on the job queue.

***YES** The job is held when it is put on the job queue until it is released ended.

Top

Schedule date (SCDDATE)

Specifies the date on which the scheduled job is submitted to the job queue and is eligible to run.

If your system or your job is configured to use the Julian date format, *MONTHSTR and *MONTHEND are calculated as if the system or job did not use the Julian date format.

***CURRENT**
The submitted job becomes eligible to run on the current date.

***MONTHSTR**
The submitted job becomes eligible to run on the first day of the month. If today is the first day of the month and the time you specify on the SCDTIME parameter has not passed, the job becomes eligible to run today. Otherwise, the job becomes eligible on the first day of the next month.

***MONTHEND**
The submitted job becomes eligible to run on the last day of the month. If today is the last day of the month and the time you specify on the SCDTIME parameter has not passed, the job becomes eligible to run today. Otherwise, if today is the last day of the month and the time you specify on the SCDTIME parameter has passed, the job becomes eligible on the last day of the next month.

***MON**
The job becomes eligible to run on Monday.

***TUE** The job becomes eligible to run on Tuesday.

***WED** The job becomes eligible to run on Wednesday.

***THU** The job becomes eligible to run on Thursday.

***FRI** The job becomes eligible to run on Friday.

***SAT** The job becomes eligible to run on Saturday.

***SUN** The job becomes eligible to run on Sunday.

date Specify a date in the job date format with or without separators.

Top

Schedule time (SCDTIME)

Specifies the time on the scheduled date at which the job becomes eligible to run.

Note: Although the time can be specified to the second, the load on the system may affect the exact time at which the submitted job becomes eligible to run.

The order that job entries with identical SCDDATE and SCDTIME values appear on the job queue may be different than the order in which they arrived. Likewise, these jobs may leave the job queue to be processed in an order different than the order in which they were entered. Do not assume jobs are entered or processed sequentially when they are scheduled to start at exactly the same time.

***CURRENT**
The job is submitted at the current time.

time Specify the time you want the job to start. 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 digits 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.

Top

Job date (DATE)

Specifies the date that is assigned to the job when it is started.

***JOBD**
The date specified in the job description is the job date.

***SYSVAL**
The value in the QDATE system value at the time the job is started is the job date.

date Specify the value that is used as the job date when the job is started; the date must be specified in the job date format.

Top

Job switches (SWS)

Specifies the first settings for a group of eight job switches used with this job. These switches can be set or tested in a CL program and used to control the flow of the program. Only 0’s (off) and 1’s (on) can be specified in the 8-digit character string.

***JOBD**
The value specified in the job description is the first setting for the job’s switches.

character-value
Specify any combination of eight zeros and ones that is used as the first switch setting for the submitted job.

Top

Allow display by WRKSBMJOB (DSPSBMJOB)

Specifies whether the job being submitted is allowed to be shown on the **Work with Submitted Jobs** panel. Any submitted job of the type specified by the **Jobs submitted from (SBMFROM)** parameter of the Work with Submitted Jobs (WRKSBMJOB) command can be shown if the job is not prevented by this parameter.

***YES** This job can be shown by the WRKSBMJOB command.

***NO** This job is not shown on any display produced by the WRKSBMJOB command.

Top

Submitted for (SBMFOR)

Specifies the job name to be used on the **Jobs submitted from (SBMFROM)** parameter of the Work with Submitted Jobs (WRKSBMJOB) command.

Single values

***CURRENT**
The name of the currently active job is used.

Qualifier 1: Submitted for

name Specify the name of the job.

Qualifier 2: User

name Specify the user name that identifies the user profile under which the job is run.

Qualifier 3: Number

000000-999999
Specify the job number.

Note: You must have job control (*JOBCTL) special authority to use this parameter.

Top

Message queue (MSGQ)

Specifies the message queue to which a completion message is sent when the submitted job has completed running, either normally or abnormally. If an abnormal ending occurs, the help information for the completion message specifies the possible causes.

Single values

***USRPRF**
A completion message is sent to the message queue specified in the user profile of the submitter.

***WRKSTN**
A completion message is sent to the work station message queue of the work station from which the job was submitted. If the job is submitted by a batch job, no completion message is sent.

***NONE**
No completion message is sent.

Qualifier 1: Message queue

name Specify the name of the message queue to which the completion message is 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 used to locate the message queue. If no library is specified as the current library for the thread, the QGPL library is used.

name Specify the library where the message queue is located.

Top

Sort sequence (SRTSEQ)

Specifies the sort sequence table to be used for string comparisons for this job.

Single values

***CURRENT**
The sort table specified for the job that is currently running is used.

***USRPRF**
The sort table specified in the user profile under which the submitted job initially runs is used. The user profile is specified on the **User (USER)** parameter.

***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.

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 library where the sort sequence table is located.

Top

Language ID (LANGID)

Specifies the language identifier to be associated with this job. The language identifier is used when *LANGIDUNQ or *LANGIDSHR is specified on the **Sort sequence (SRTSEQ)** parameter. If the job CCSID is 65535, this parameter is also used to determine the value of the job default CCSID (DFTCCSID).

***CURRENT**
The language identifier specified for the job that is currently running is used.

***USRPRF**
The language ID specified in the user profile under which the submitted job runs is used. The user profile is specified on the **User (USER)** parameter.

***SYSVAL**
The system value QLANGID is used.

character-value
Specify the language identifier to be used by the job.

Top

Country or region ID (CNTRYID)

Specifies the country or region identifier (ID) to be used by the job.

***CURRENT**
The country or region ID specified for the job that is currently running is used.

***USRPRF**
The country or region ID specified in the user profile under which the submitted job runs is used. The user profile is specified on the **User (USER)** parameter.

***SYSVAL**
The system value QCNTRYID is used.

character-value
Specify the country or region ID to be used by the job.

Top

Coded character set ID (CCSID)

Specifies the coded character set identifier (CCSID) used for the submitted job.

***CURRENT**
The CCSID specified for the job that is currently running is used.

***USRPRF**
The CCSID specified in the user profile where the submitted job initially runs is used.

***SYSVAL**
The CCSID specified in the system value QCCSID at the time the job is started is used.

***HEX** The CCSID 65535 is used.

1-65535
Specify the CCSID.

Top

Copy environment variables (CPYENVVAR)

Specifies whether the environment variables from the submitting job are copied to the new job.

***NO** The environment variables are not copied.

***YES** The environment variables are copied.

Top

Allow multiple threads (ALWMLTTHD)

Specifies whether or not the job is allowed to run with multiple user threads. This attribute does not prevent the operating system from creating system threads in the job. This attribute is not allowed to be changed after the job is submitted.

***JOBD**
The value specified in the job description determines whether or not the job is allowed to run with multiple user threads.

***NO** The job is not allowed to run with multiple user threads.

***YES** The job is allowed to run with multiple user threads.

Top

Spooled file action (SPLFACN)

Specifies whether or not spooled files are accessed through job interfaces after the job ends. Keeping spooled files with jobs allows job commands such as Work with Submitted Jobs (WRKSBMJOB) to work with the spooled files even after the job has ended. Detaching spooled files from jobs reduces the use of system resources by allowing job structures to be recycled when the jobs end.

***CURRENT**
The value from the current job is used for the submitted job.

***JOBD**
The value in the job description is used.

***SYSVAL**
The value specified in the system value QSPLFACN is used.

***KEEP**
When the job ends, if spooled files for the job exist in the system auxiliary storage pool (ASP 1) or in a basic user ASP (ASPs 2-32), the spooled files are kept with the job and the status of the job is updated to indicate that the job has completed. When all remaining spooled files for the job are in independent ASPs (ASPs 33-255), the spooled files will be detached from the job and the job will be removed from the system.

***DETACH**
When the job ends, the spooled files are detached from the job and the job is removed from the system.

Top

Examples

Example 1: Submitting a Job

```
SBMJOB   JOB(SPECIAL)  JOBD(MYLIB/MYJOBD)  CMD(CALL MYPROG)
```

This command causes the job named SPECIAL to be submitted. Most of the attributes for the job are taken from the job description MYJOBD, or the job that is currently running, except for the command. The CALL command is placed on the submitted job's message queue so that the program MYPROG can be called and run later.

Example 2: Submitting a Job

```
SBMJOB   JOB(PAYROLL)  JOBD(PAYROLL)  INQMSGRPY(*RQD)
```

This command submits a job named PAYROLL to the system. All the information needed for this job (such as the job queue and routing data but not the inquiry message control value) is contained in the job description PAYROLL, or the job that is currently running. The library list in effect for the job issuing this command is used to find the job description. All inquiry messages sent during running of this job requires the receiver of the inquiry message to reply.

Example 3: Submitting a Job to a Job Queue

```
SBMJOB   JOBD(*USRPRF)  JOB(COPY12)  JOBQ(NIGHTQ)
         CMD(CPYF FILEA FILEB)
```

This command submits the job COPY12, which uses the job description in the user profile of the submitting job, to the job queue NIGHTQ. The CMD parameter provides the CL command necessary for the job to run. A command such as this might be used to copy the file at night while the system is unattended.

Top

Error messages

***ESCAPE Messages**

CPF133A
SBMJOB not allowed during IPL.

CPF1338
Errors occurred on SBMJOB command.

CPF1651
Sort sequence table not accessed.

Top

Submit Network Job (SBMNETJOB)

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

Parameters
Examples
Error messages

The Submit Network Job (SBMNETJOB) command sends an input stream to another user on the SNADS network. (The input stream is sent to another user where it can be filed, submitted, or rejected.) When the input stream arrives, its placement is governed by the job action (JOBACN) network attribute. If the value of JOBACN is *SEARCH, the entry in the network job table at the receiving system is used to determine the action taken. At the receiving system, the job may be submitted immediately, filed for placement by the receiving user, or rejected.

When the input stream arrives at the destination system, a message is sent to both the recipient of the input stream as well as the originator of the input stream stating that the input stream arrived. This command can only be used to send a batch input stream to a user on a remote system.

Restrictions:

1. To use this command, the user must have object operational and read authority to the file that is submitted, and for the library that contains the file.
2. The user must be enrolled in the system distribution directory to use this command. (For information on enrolling in the system distribution directory, see the SNA Distribution Services book, SC41-5410.
3. If the job action (JOBACN) network attribute on the receiving system is set to *SEARCH, there must be an entry for the user in the network job table on the receiving system. The entry in this table specifies a user profile on the receiving system that is used to verify that the user is authorized to submit the job on that system. The user profile on the receiving system must be authorized to use the job queues, and must have object operational authority for the job descriptions specified by the JOB commands in the input stream.
4. The file that is submitted cannot contain more than approximately 2 billion bytes of data.

Top

Parameters

Keyword	Description	Choices	Notes
FILE	File	*Qualified object name*	Required, Positional 1
	Qualifier 1: File	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
TOUSRID	User ID	Values (up to 50 repetitions): *Element list*	Required, Positional 2
	Element 1: User ID	*Character value*	
	Element 2: Address	*Character value*	
MBR	Member	*Name*, ***FIRST**	Optional, Positional 3
PTY	Send priority	***NORMAL**, *HIGH	Optional

Top

File (FILE)

Specifies the name and library of the physical file containing the input stream that is sent.

This is a required parameter.

The possible library values are:

***LIBL** The library list 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, QGPL is used.

library-name
Specify the library where the file is located.

Top

User ID (TOUSRID)

Specifies the two-part user ID of one or more users to whom the input stream is submitted, or the name of one or more distribution lists containing the two-part user IDs of one or more users to whom the file is to be sent. A combination of both user IDs and distribution lists can be specified on the same command. Each user ID or distribution list is specified as a two-part name, and both parts are required.

NOTES:

1. Depending on the type of work station being used, the internal value for a user identifier may differ from the characters shown by the Display Directory Entries (DSPDIRE) command. If the byte-string value specified for the TOUSRID parameter does not match the rules for an internal user identifier value, or if it does not match the internal value for any enrolled user, an error may be reported.
2. The user specified in this parameter, or in the distribution list, must be a remote user. The SBMNETJOB command cannot be used to send input streams to local users.

You can enter multiple values for this parameter.

Top

Member (MBR)

Specifies the member that is sent from the file.

***FIRST**
The first member (in creation order) in the file is submitted.

member-name
Specify the name of the file member that is submitted.

Top

Send priority (PTY)

Specifies the queuing priority used for the input stream when it is being routed through a SNADS network.

The possible values are:

***NORMAL**

The input stream is sent with a service level priority of data low, which is used for most data traffic. On a System i5, data low distributions are placed on the normal distribution queue specified for the route.

***HIGH**

The input stream is sent with a service level priority of data high, which is used for high priority data traffic. On a System i5, data high distributions are placed on the data high distribution queue specified for the route.

Top

Examples

```
SBMNETJOB   FILE(PAYROLL)  TOUSRID(PAYROLL SYSTEM1)
            MBR(WEEKLY)
```

This command sends the input streams contained in member WEEKLY of file PAYROLL to user ID PAYROLL SYSTEM1.

Top

Error messages

***ESCAPE Messages**

CPF8056
File &1 in &2 not a physical file.

CPF8058
File &1 is a spooled file.

CPF8063
Cannot assign necessary resource.

CPF8065
Input stream &1 in &2 member &3 not sent to any users.

CPF8066
One or more user identifiers on this command is not correct.

CPF8068
Error detected while processing file to be sent.

CPF8072
Object to be sent is greater than maximum size of 2GB.

CPF9005
System resource required to complete this request not available.

CPF9006
User not enrolled in system distribution directory.

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.

CPF9812
File &1 in library &2 not found.

CPF9820
Not authorized to use library &1.

CPF9822
Not authorized to file &1 in library &2.

CPF9830
Cannot assign library &1.

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.

CPF9848
Cannot open file &1 in library &2 member &3.

CPF9849
Error while processing file &1 in library &2 member &3.

Top

Submit Network Server Command (SBMNWSCMD)

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

Parameters
Examples
Error messages

The Submit Network Server Command (SBMNWSCMD) command submits a command to run on the designated server. For SVRTYPE(*WINDOWS) or (*WINDOWSNT), command output is directed as specified by the CMDSTDOUT parameter.

Top

Parameters

Keyword	Description	Choices	Notes
CMD	Command	*Character value*, *NOLOGCMD	Required, Positional 1
SERVER	Server	*Character value*	Required, Positional 2
NOLOGCMD	Command (not logged)	*Character value*	Optional
SVRTYPE	Server type	***NWSUSRA**, *NWSA, *WINDOWS, *WINDOWSNT	Optional
CMDTYPE	Command type	***SVRTYPE**, *WINDOWS, *WINDOWSNT	Optional
LODUSRPRF	Load user profile	***YES**, *NO	Optional
AUTDMN	Authentication domain	*Character value*, ***PRIMARY**, *LOCAL	Optional
CMDSTDOUT	Command standard output	*Path name*, ***JOBLOG**, *PRINT	Optional
CVTSTDOUT	Convert standard output	***YES**, *NO	Optional

Top

Command (CMD)

Specifies the command submitted to the network server. The syntax of the command string specified for this parameter is not checked.

***NOLOGCMD**
Specifies the user wants to submit a command string that is not logged in the joblog. This is useful if the command string contains sensitive data such as passwords. If *NOLOGCMD is specified, a command string must be entered on the NOLOGCMD parameter.

command
The command string that is to be submitted to the network server. This command string will be shown in the joblog.

This is a required parameter.

Top

Server (SERVER)

Specifies the name of the server to which the command is submitted.

This is a required parameter.

Top

Command (not logged) (NOLOGCMD)

Specifies the command string to be submitted to the network server. This command string will not be echoed to the joblog. Use this parameter to submit commands that contain sensitive data such as passwords. This parameter is required if, and is only allowed when, CMD(*NOLOGCMD) is specified.

Note: Since the command is being executed on a server, the presence of this option does not prevent the server from returning the command string as part of the output data. Thus, depending on the command, it may still be returned and displayed in the job log or spool file.

Top

Server type (SVRTYPE)

Specifies the type of server to which the command is sent.

***NWSUSRA**
The server type specified in the network server attributes (CHGNWSUSRA command) for the user profile running the SBMNWSCMD is used.

***NWSA**
The server type specified in the network server attributes (CHGNWSA command) is used.

***WINDOWS or *WINDOWSNT**
The server type is Windows. Only Windows commands will run on a local Windows server.

Note: *WINDOWS should be used in V5R4 and later releases. The *WINDOWSNT value is supported for compatibility with releases prior to V5R4.

Top

Command type (CMDTYPE)

Specifies the type of command string specified in the CMD parameter. This is used by the system to determine what type of processing needs to occur for the command string.

Note: This parameter is ignored.

***SVRTYPE**
The command string is processed based on the type of server specified in the SVRTYPE parameter. If the server is a *WINDOWS or a *WINDOWSNT server, the command string is assumed to be a Windows command.

***WINDOWS or *WINDOWSNT**
The command string is a Windows command.

Note: *WINDOWS should be used in V5R4 and later releases. The *WINDOWSNT value is supported for compatibility with releases prior to V5R4.

Top

Load user profile (LODUSRPRF)

Specifies whether or not to load the user profile on the remote server as part of the login environment.

Note: This parameter is only valid when SVRTYPE(*WINDOWS) or (*WINDOWSNT) is specified.

***YES** Load the user profile.

***NO** Do not load the user profile.

Top

Authentication domain (AUTDMN)

Specifies the Windows domain where the user is authenticated.

Note: This parameter is only valid when SVRTYPE(*WINDOWS) or (*WINDOWSNT) is specified.

***PRIMARY**
The user is authenticated on the primary domain of the server.

***LOCAL**
The user is authenticated on the local server.

'domain-name'
Specify the domain name where the user is authenticated.

Top

Command standard output (CMDSTDOUT)

Specifies where the standard output returned from the command is to be stored, if any exists. Standard output can be written to the job log of the job that issues this command, it can be written to a spooled file, or it can be written to a file. The standard error returned from the command is always directed to the job log of the job that issues this command, if any exists.

Note: This parameter is only valid when SVRTYPE(*WINDOWS) or (*WINDOWSNT) is specified.

***JOBLOG**
The standard output of the network server command is directed to the job log of the job that issues this command. It shares the job log with the standard error output of the network server command. Both may be mixed in the job log, depending on the order by which the command writes standard output and standard error information.

***PRINT**
The standard output of the network server command is directed to a spooled file. If CVTSTDOUT(*YES) is specified, certain control characters such as line feeds and carriage returns are converted to new lines and other non-displayable control characters such as highlight and underscore are converted to blanks.

'stream-file-path-name'
Specify the path name of the stream file to which the standard output of the network server command is directed. The specified path must exist. If the stream file doesn't exist, it is created. If the stream file exists, all data is overlaid. Additional information about path name is in the Integrated file system topic collection in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/ book.

Top

Convert standard output (CVTSTDOUT)

Specifies whether the standard output will be converted from the server's code set to the CCSID of the i5/OS user profile that submitted the command. For binary output, CVTSTDOUT(*NO) should be specified.

Note: This parameter is only valid when CMDSTDOUT(*PRINT) or CMDSTDOUT(stream-file-path-name) is specified.

<u>*YES</u> The output is converted from the server's code set to the CCSID of the i5/OS user profile that submitted the command.

***NO** The output is not converted from the server's code set.

Top

Examples

Example 1: Submitting a Windows Command

```
SBMNWSCMD   CMD('net config server')  SERVER(WINSVR)
            CMDSTDOUT(*JOBLOG)
```

This command will display the Windows Server service settings on the Windows server WINSVR. Standard output from the command is returned to i5/OS and directed to the job log.

Top

Error messages

<u>*ESCAPE Messages</u>

CPFA43F
Network server command not submitted.

CPFA46C
Unable to complete command processing on server &1.

CPFA46F
Network server description &1 not found.

Top

Submit Remote Command (SBMRMTCMD)

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

Parameters
Examples
Error messages

The Submit Remote Command (SBMRMTCMD) command sends a CL command through Distributed Data Management (DDM) to be run on the target system specified by a DDM file. The DDM file determines the communications line used, and indirectly identifies the target system that receives the submitted command.

This command sends only CL commands to a remote system which supports the SBMRMTCMD command language. It cannot send non-CL commands to a different system type (for example, OCL commands cannot be sent to a target System/36).

The primary purpose of this command is to allow a source system user or program to perform file management operations and file authorization activities on files located on a target system.

- Create or delete physical, logical, device, or source files
- Grant or revoke object authority to remote files
- Check, rename, or move files or other objects
- Save or restore files or other objects

Restrictions:

- File processing on the target system operates independently of processing on the source system. Commands dependent on a specific recursion level or request level may *not* function as expected.
- Output (such as spooled files) generated by a submitted command exists only on the target system. The output is *not* sent back to the source system. Therefore display commands or commands that are used to service programs should not be sent because the output results remain at the target system.
- Translation is not performed for *impromptu* messages caused by target system errors, because they are not stored on the system; the text for an impromptu message is sent directly to the source system and displayed. The message identifier of all other message types generated on the remote system is sent back to the source system. The message text that exists for the message identifier on the source system is displayed as it would be for the same error on the source system.
- A maximum of 10 messages, generated during the running of a submitted command, can be sent by the target system to the source system. If more than 10 messages are generated, an additional *informational* message is sent that indicates that the messages exist in the job log for the target job on the target system. If one of those messages is an *escape* message, the first nine messages of other types are sent, followed by the informational message and the escape message.
- In multithreaded jobs, this command is not threadsafe and fails for DDM files of type *SNA.

Top

Parameters

Keyword	Description	Choices	Notes
CMD	Command to run	*Character value*	Required, Positional 1

Keyword	Description	Choices	Notes
DDMFILE	DDM file	*Qualified object name*	Required, Positional 2
	Qualifier 1: DDM file	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	

Top

Command to run (CMD)

Specifies a character string of up to 2000 characters that represents a command that is run on the target system. The command must be allowed in both batch and interactive environments on the target system. The command must be enclosed in apostrophes if it contains embedded blanks or special characters. The command must use the syntax of the target system.

This is a required parameter.

Note: The normal rule of pairing apostrophes in quoted strings on the local system must be *doubled* when the same string is submitted to a remote system. This is required to allow coding a quoted string within another quoted string.

Top

DDM file (DDMFILE)

Specifies the name and library of the Distributed Data Management (DDM) file that is used to submit the command to the target system. The DDM file is used only to determine the remote location representing the target system. The remote file name associated with the DDM file is ignored by this command.

This is a required parameter.

The possible library values are:

***LIBL** The library list 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 library where the file is located.

Top

Examples

Example 1: Deleting a File

```
SBMRMTCMD   CMD('DLTF LIBX/FRED')  DDMFILE(DENVER)
```

This command deletes the file named FRED in library LIBX on the target system that is associated with the DDM file named DENVER.

Example 2: Creating a Physical File

```
SBMRMTCMD   CMD('CRTPF  SALES/CAR  QGPL/QDDSSRC MASTER')
            DDMFILE(DENVER)
```

This command creates the physical file CAR in library SALES using the data description specifications (DDS) in the source file QDDSSRC and source member named MASTER in the QGPL library. The DDS must already exist on the target system identified by the DDM file named DENVER in the target job's library list.

Example 3: Changing the Text Description

```
SBMRMTCMD   CMD('CHGDDMF  FILE(LIBX/STANLEY)
              TEXT(''Don''''t forget to pair apostrophes.'')')
            DDMFILE(SMITH)
```

This command changes the text in the description of the DDM file named STANLEY which is stored in library LIBX. Because the submitted command requires an outside set of single apostrophes (for the CMD parameter), each single or double apostrophe normally required in the TEXT parameter for *local* system processing must be doubled for *remote* system processing. The coding above produces a single apostrophe in the text when it is shown or printed on the remote system.

Example 4: Creating a DDM File

```
SBMRMTCMD   CMD('CRTDDMF FILE(SALES/MONTHLY)
                RMTFILE(*NONSTD ''CAR.SALES(JULY)''
                RMTLOCNAME(DALLAS)')
            DDMFILE(CHICAGO)
```

This command creates (on the target system identified by the information in the DDM file named CHICAGO) another DDM file named MONTHLY. The new DDM file is stored in a library named SALES on the CHICAGO system. The new DDM file on the CHICAGO system is used to access a file and *member* on a different system named DALLAS. The accessed file is named SALES/CAR and the member name in the file is JULY.

Note that this CRTDDMF command string contains *three* sets of single apostrophes: one set to enclose the entire command being submitted, and a double set to enclose the file and member named in the RMTFILE parameter. This is how any i5/OS file *member* name must be specified on the SBMRMTCMD command, because of the parentheses needed to enclose the member name.

Example 5: Replacing a Portion of the Library List

```
SBMRMTCMD   CMD('RPLLIBL  LIBL(QGPL QTEMP SALES EVANS)')
            DDMFILE(EVANS)
```

This command replaces the user's portion of the library list being used by the target job associated with the DDM file named EVANS, which is being used by the source job in which this SBMRMTCMD command is being submitted. In that source job, if there are other open DDM files that specify the same device and mode, this library list is used for them also.

Top

Error messages

***ESCAPE Messages**

CPF9164
Target system does not support SBMRMTCMD.

CPF9165
File &1 in library &2 not a DDM File.

CPF917A
Error occurred on distributed file.

CPF917B
Target system &3 not available.

CPF9172
SBMRMTCMD command ended abnormally.

CPF9174
Error on call to user exit program on target system.

CPF9175
Error during processing of user exit program.

CPF9177
User exit program did not complete successfully.

CPF9178
Processing of the command specified by SBMRMTCMD failed.

CPF9182
Cannot start DDM communications.

Top

Select (SELECT)

Where allowed to run:

- Batch program (*BPGM)
- Interactive program (*IPGM)

Threadsafe: Yes

Parameters
Examples
Error messages

The Select (SELECT) command begins a control structure to allow conditional processing of command groups. The select group is ended with an End Select (ENDSELECT) command and must contain one or more When (WHEN) commands and optionally an otherwise (OTHERWISE) command.

When a Select command is entered, the condition specified on the first When command is evaluated. If the condition is true, the command specified on the THEN parameter is run. After the command or group of commands is processed, control will pass to the matching ENDSELECT command. Only one WHEN or the optional OTHERWISE will be processed within a SELECT group.

If the first WHEN condition is not true, subsequent WHEN commands are evaluated in the order specified. If no WHEN command condition is true, control will pass to the command specified on the OTHERWISE command, if OTHERWISE is specified.

Restrictions:

- This command is valid only within a CL program or ILE CL procedure.
- Up to 25 levels of nested SELECT commands are allowed.

There are no parameters for this command.

Top

Parameters

None

Top

Examples

```
DCL   VAR(&INT)  TYPE(*INT)  LEN(4)
 :
SELECT
  WHEN   COND(&INT *LT 0)  THEN(DO)
    :   (group of CL commands)
  ENDDO
  WHEN   COND(&INT *EQ 0)    /* Do nothing when zero */
  WHEN   COND(&INT *GT 0)  THEN(CHGVAR &INT (&INT - 1)
ENDSELECT
```

The SELECT command group will evaluates the conditions of the WHEN commands in the order they are encountered. When a COND parameter results in a true value, the command specified on the THEN parameter is processed. After the command on the THEN parameter is processed, control passes to the command following the ENDSELECT command.

Top

Error messages

None

Top

Set ASP Group (SETASPGRP)

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

Parameters
Examples
Error messages

The Set Auxiliary Storage Pool Group (SETASPGRP) command sets the auxiliary storage pool (ASP) group for the current thread. Additionally, this command allows you to change the libraries in the library list for the current thread. If an ASP group had already been set, this command will remove the old ASP group from the current thread and set the specified ASP group for the current thread. Once the specified ASP group has been set for the current thread, all libraries in the independent ASPs in the ASP group are accessible and objects in those libraries can be referenced using regular library-qualified object name syntax. The libraries in the independent ASPs in the specified ASP group plus the libraries in the system ASP (ASP number 1) and basic user ASPs (ASP numbers 2-32) form the *library name space* for the thread. All libraries in the library list need to be in the new library name space or the library list is not changed and the new ASP group is not set.

Restrictions:

- You must have use (*USE) authority to all ASP device descriptions in the ASP group and to all the specified libraries in the library list before the library name space and the library list are changed. If you are not authorized to an ASP device description or to one of the libraries, the ASP group will not be set and the library list will not be changed.
- When *CURUSR is specified for the **ASP group (ASPGRP)** or **Libraries for current thread (USRLIBL)** parameter, you must have read (*READ) authority to the job description listed in your user profile and execute (*EXECUTE) authority to the library where the job description is located.
- The SETASPGRP command is not allowed in the following:
  - System jobs QPFRADJ, QJOBSCD, QSYSARB, QSYSARB3, QSYSARB4 and QLUS.
  - All subsystem monitor jobs.
  - DDM, DRDA, database host server and SQL server jobs once the initial namespace has been established.
  - Receive Journal Entry (RCVJRNE) and Delete Journal Receiver (DLTJRNRCV) command exit programs.
  - Management Central Registered Inventory Gathering Service (RIGS) Exit Programs (exit point QIBM_QYIV_INVGTRSRV).
  - Trigger or format selector programs that run as part of database I/O operations.
  - Attention programs (the PGM parameter of the SETATNPGM command).
  - Break handling programs (the PGM parameter of the CHGMSGQ command).
  - Programming Development Manager (PDM) functions.

Top

Parameters

Keyword	Description	Choices	Notes
ASPGRP	ASP group	*Name*, *CURUSR, *NONE	Required, Positional 1
SYSLIBL	System library list	***CURSYSBAS**, *SYSVAL	Optional
CURLIB	Current library	*Name*, ***CURSYSBAS**, *CURUSR, *CRTDFT	Optional

Keyword	Description	Choices	Notes
USRLIBL	Libraries for current thread	Single values: ***CURSYSBAS**, *CURUSR, *SYSVAL, *NONE Other values (up to 250 repetitions): *Name*	Optional

Top

ASP group (ASPGRP)

Specifies the name of the auxiliary storage pool (ASP) group to set for the current thread. The ASP group name is the name of the *primary* ASP device within the ASP group.

This is a required parameter.

***CURUSR**
The ASP group is set to the value defined for the **Initial ASP group (INLASPGRP)** in the default job description of the user profile that the thread is currently running under.

***NONE**
Specify for the current thread to have no ASP group. The library name space will not include libraries from any ASP group. Only the libraries in the system ASP and any basic user ASPs will be in the library name space.

name Specify the name of the *primary* ASP in the ASP group to be set for the current thread. All libraries from all ASPs in this ASP group will be included in the library name space.

Top

System library list (SYSLIBL)

Specifies the system part of the library list for the thread in which the command is entered.

***CURSYSBAS**
The libraries in the system part of the library list of the current thread that are found in the system ASP (ASP number 1) or any configured basic user ASP (ASP numbers 2-32) are used as the new system part of the library list.

***SYSVAL**
The system part of the library list of the current thread is set from the current value of system value QSYSLIBL.

Top

Current library (CURLIB)

Specifies the library to be used in the current library entry of the library list for the thread. If *CURUSR or a library name is specified and the library cannot be found in the new library name space, an error message is sent and the library list and ASP group are not changed.

***CURSYSBAS**
The library name in the current library entry of the library list is used as the new current library if the library is found in the system ASP (ASP number 1) or any configured basic user ASP (ASP numbers 2-32). If the library name in the current entry is not found in the system ASP or any basic user ASP, the current library entry will be removed from the library list.

***CURUSR**
The current library is set to the value defined for the **Current library (CURLIB)** attribute of the user profile that the thread is currently running under.

***CRTDFT**
Changes the library list to remove any name from the current library entry. If objects are created specifying *CURLIB for the library name, library QGPL will be used.

name Specify the name of the library that replaces the current library entry in the library list.

Top

Libraries for current thread (USRLIBL)

Specifies the libraries that are placed in the user part of the library list. If *CURUSR or a list of library names is specified and any of these libraries cannot be found in the new library name space, an error message is sent and the library list and ASP group are not changed.

Single values

***CURSYSBAS**
The libraries in the user part of the library list of the current thread that are found in the system ASP (ASP number 1) or any configured basic user ASP (ASP numbers 2-32) are used as the new user part of the library list.

***CURUSR**
The user part of the library list for the thread is set to the value defined for the **Initial library list (INLLIBL)** in the default job description of the user profile that the thread is currently running under.

***SYSVAL**
The user part of the library list is set from the current value of system value QUSRLIBL.

***NONE**
Changes the user part of the library list to remove all library names.

Other values (up to 250 repetitions)

name Specify the names of the libraries to be used as the user part of the library list, in the order in which they are to be searched.

Top

Examples

Example 1: Set New ASP Group

```
SETASPGRP   ASPGRP(WAREHOUSE1) SYSLIBL(*SYSVAL)  CURLIB(*CURUSR)
            USRLIBL(*CURSYSBAS)
```

This command will set the auxiliary storage pool (ASP) group for the thread in which the command runs to be WAREHOUSE1. This will change the library name space for the thread to include all libraries in any of the independent ASPs in the ASP group identified by the independent ASP device named WAREHOUSE1. The system part of the library list will be set from system value QSYSLIBL. The current library entry of the library list will be set from the *Current library* value defined in the user profile that the thread is currently running under. The user part of the library list will be set using the current user part of the library list and removing any libraries that are not found in the system ASP or configured basic user ASPs.

Example 2: Set to No ASP Group

```
SETASPGRP   ASPGRP(*NONE)  SYSLIBL(*CURSYSBAS)  CURLIB(*CRTDFT)
            USRLIBL(*NONE)
```

This command will remove any ASP group for the thread in which the command runs. This will change the library name space for the thread to include only those libraries in the system ASP (ASP number 1) and basic user ASPs (ASP numbers 2-32). The system part of the library list will be set using the current system part of the library list and removing any libraries that are not found in the system ASP or configured basic user ASPs. The current library entry of the library list will be changed to be empty which will cause library QGPL to be used as the current library. The user part of the library list will be changed to be empty.

Top

Error messages

***ESCAPE Messages**

CPFB8E9
ASP group &1 not set for thread &2.

Top

Set Attention Program (SETATNPGM)

Where allowed to run: Interactive environments (*INTERACT *IPGM *IREXX *EXEC)
Threadsafe: No

Parameters
Examples
Error messages

The Set Attention Program (SETATNPGM) command sets up a program that is called when the Attention key is pressed. The setting is in effect for this recursion level and lower levels if more programs are called, but it is no longer in effect if the job returns from this recursion level to the previous one. If the Attention key handler's status is on, the specified program is called when the key is pressed. No parameters are passed to the Attention key handler when it is called. The Attention handling program runs in the same process with the same job attributes, overrides, and group authorities as the program that issued the SETATNPGM command. However, program adopted authority is not carried over.

The handling of the attention key does not change the auxiliary storage pool (ASP) group that is in effect. The ASP group determines the library name space of the thread. If the attention handling program resides in an independent ASP and if that ASP is not part of the library name space for the thread at the time the attention key is handled, the intended program will not be found. To avoid potential problems when applications change the library name space, the program should reside in the system ASP or a basic user ASP.

Top

Parameters

Keyword	Description	Choices	Notes
PGM	Program	Single values: *CURRENT, *PRVINVLVL Other values: *Qualified object name*	Required, Positional 1
	Qualifier 1: Program	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
SET	Set attention key	***ON**, *OFF	Optional, Positional 2

Top

Program (PGM)

Specifies the qualified name of the program to be the Attention key handler at this recursion level.

This is a required parameter.

Single values

***CURRENT**
The program name of the Attention key handler currently in effect is used as the value of this parameter.

***PRVINVLVL**
The Attention key handler in effect at the previous recursion level is reinstated as the Attention key handler at this recursion level. The **Set attention key (SET)** parameter is not allowed if this

special value is specified, because the SET status of the previous recursion level is also reinstated. This option is used when a program has specified an Attention program and wants to revert back to a previous level.

Qualifier 1: Program

name Specify the name of the Attention key handler program.

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 attention program is located.

Top

Set attention key (SET)

Specifies whether the Attention key handler indicated in the **Program (PGM)** parameter is called when the Attention key is pressed. This parameter is not allowed if *PRVINVLVL is specified for the PGM parameter.

***ON** The Attention key handler specified in the **Program** (PGM) parameter is called when the Attention key is pressed.

***OFF** The Attention key handler specified in the **Program** (PGM) parameter is not called when the Attention key is pressed.

Top

Examples

Example 1: Setting the ATTN Key Handler

```
SETATNPGM   PGM(QGPL/ATTN)  SET(*ON)
```

This command causes the program QGPL/ATTN to become the ATTN key handler. Because SET(*ON) is specified, the program is called when the ATTN key is pressed.

Example 2: Setting the Attention Key Off

```
SETATNPGM   PGM(*CURRENT)  SET(*OFF)
```

The current attention handling program has its status changed to SET(*OFF). Because the status is SET(*OFF) when the ATTN key is pressed, the attention handling program is not called.

Example 3: Previous Recursion-Level Support

```
SETATNPGM   PGM(*PRVINVLVL)
```

The attention handling program and status that was in effect at the previous recursion level is reinstated at this recursion level. If no attention handler is in effect, after this command is run nothing happens when the ATTN key is pressed.

Example 4: Emulating the System Request Key

```
SETATNPGM   PGM(QWSSYSRQ)
```

The system-supplied program QWSSYSRQ will be called when the ATTN key is pressed. This system program allows the ATTN key to act as a system request key by showing the system request menu on the display when the ATTN key is pressed.

Top

Error messages

***ESCAPE Messages**

CPF1318

Attention key program &1 in &2 not set.

Top

Set Customization Data (SETCSTDTA)

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

Parameters
Examples
Error messages

The Set Customization Data (SETCSTDTA) command can be used to copy and delete information in order to customize data, such as work area content and mouse double-click action values, for Graphical Operations support.

Using this command, administrators can set up users by getting information from a named user profile and copying it to other user profiles. Administrators can also remove the customized values for a named user, and choose to replace those customized values with data from a named user profile.

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 SETCSTDTA

***ESCAPE Messages**

CPF2204
User profile &1 not found.

CPF2217
Not authorized to user profile &1.

GUI0085
&1 does not have customization data.

GUI0087
Some user profiles did not have customization data copied successfully.

GUI0091
Some user profiles did not have customization data deleted successfully.

GUI0117
&1 is not valid for subset key.

Top

Parameters

Keyword	Description	Choices	Notes
USRPRF	User profile	Values (up to 300 repetitions): *Simple name*	Required, Positional 1
VALUE	Customization data to use	*Simple name*, *NONE	Required, Positional 2
REPLACE	Replace customization data	***NO**, *YES	Optional, Positional 3

Top

User profile (USRPRF)

Specifies the names of the user profiles for which customized values are to be set or removed with the option to set them again. A maximum of 300 user profiles can be named.

You can enter multiple values for this parameter. If you are on an entry display and you need additional entry fields to enter these multiple values, type a plus sign (+) in the entry field opposite the phrase ″+ for more″, and press the Enter key.

This is a required parameter.

Top

Customization data to use (VALUE)

Specifies the customization data value to which the user profile named on the **User profile** prompt (USRPRF parameter) is to be set.

The possible values are:

***NONE**
The customization data is to be removed from the user profile and no customization value is to be set.

user-profile
Specify the name of the user profile from which the customization value data is to be copied and set.

This is a required parameter.

Top

Replace customization data (REPLACE)

Specifies whether to replace existing customization data if the user profile named on the **User profile** prompt (USRPRF parameter) has customization data.

The possible values are:

***NO** The existing customization data is not replaced. An exception message is issued.

***YES** The existing customization data is replaced with the customization data from the user profile specified on the **Customization data to use** prompt (VALUE parameter).

Top

Examples

None

Top

Error messages

***ESCAPE Messages**

CPF2204
User profile &1 not found.

CPF2217
Not authorized to user profile &1.

GUI0085
&1 does not have customization data.

GUI0087
Some user profiles did not have customization data copied successfully.

GUI0091
Some user profiles did not have customization data deleted successfully.

GUI0117
&1 is not valid for subset key.

Top

Set Keyboard Map (SETKBDMAP)

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

Parameters
Examples
Error messages

The Set Keyboard Map (SETKBDMAP) command allows the user to override the PA (Program Attention) and PF (Program Function) key assignment defaults provided by the system.

This command assigns the specified F-to-PF map to the device where the command was entered (if it is a 3270 display station device) or to the 3270 display station specified if the user has authority to that device. More information on the user-assignment keyboard mapping is in Remote Work Station Support book, SC41-5402 book.

Top

Parameters

Keyword	Description	Choices	Notes
DEV	Device	_Name_, __*REQUESTER__	Optional, Positional 1
PF1	PF1 key value	__*HELP__, *ATTN, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, *F15, *F16, *F17, *F18, *F19, *F20, *F21, *F22, *F23, *F24, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional
PF2	PF2 key value	*ATTN, *HELP, __*HLP3270__, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, *F15, *F16, *F17, *F18, *F19, *F20, *F21, *F22, *F23, *F24, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional
PF3	PF3 key value	*ATTN, *HELP, *HLP3270, __*CLEAR__, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, *F15, *F16, *F17, *F18, *F19, *F20, *F21, *F22, *F23, *F24, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional
PF4	PF4 key value	*ATTN, *HELP, *HLP3270, *CLEAR, __*PRINT__, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, *F15, *F16, *F17, *F18, *F19, *F20, *F21, *F22, *F23, *F24, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional
PF5	PF5 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, __*DSPATR__, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, *F15, *F16, *F17, *F18, *F19, *F20, *F21, *F22, *F23, *F24, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional
PF6	PF6 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, __*TEST__, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, *F15, *F16, *F17, *F18, *F19, *F20, *F21, *F22, *F23, *F24, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional
PF7	PF7 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, __*DOWN__, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, *F15, *F16, *F17, *F18, *F19, *F20, *F21, *F22, *F23, *F24, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional

Keyword	Description	Choices	Notes
PF8	PF8 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, ***UP**, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, *F15, *F16, *F17, *F18, *F19, *F20, *F21, *F22, *F23, *F24, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional
PF9	PF9 key value	*NONE, ***ATTN**, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, *F15, *F16, *F17, *F18, *F19, *F20, *F21, *F22, *F23, *F24, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional
PF10	PF10 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, ***RESET**, *SYSREQ, *BCKSPC, *F13, *F14, *F15, *F16, *F17, *F18, *F19, *F20, *F21, *F22, *F23, *F24, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional
PF11	PF11 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, ***SYSREQ**, *BCKSPC, *F13, *F14, *F15, *F16, *F17, *F18, *F19, *F20, *F21, *F22, *F23, *F24, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional
PF12	PF12 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, ***BCKSPC**, *F13, *F14, *F15, *F16, *F17, *F18, *F19, *F20, *F21, *F22, *F23, *F24, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional
PF13	PF13 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, ***F13**, *F14, *F15, *F16, *F17, *F18, *F19, *F20, *F21, *F22, *F23, *F24, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional
PF14	PF14 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, ***F14**, *F15, *F16, *F17, *F18, *F19, *F20, *F21, *F22, *F23, *F24, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional
PF15	PF15 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, ***F15**, *F16, *F17, *F18, *F19, *F20, *F21, *F22, *F23, *F24, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional
PF16	PF16 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, *F15, ***F16**, *F17, *F18, *F19, *F20, *F21, *F22, *F23, *F24, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional
PF17	PF17 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, *F15, *F16, ***F17**, *F18, *F19, *F20, *F21, *F22, *F23, *F24, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional
PF18	PF18 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, *F15, *F16, *F17, ***F18**, *F19, *F20, *F21, *F22, *F23, *F24, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional
PF19	PF19 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, *F15, *F16, *F17, *F18, ***F19**, *F20, *F21, *F22, *F23, *F24, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional

Keyword	Description	Choices	Notes
PF20	PF20 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, *F15, *F16, *F17, *F18, *F19, ***F20**, *F21, *F22, *F23, *F24, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional
PF21	PF21 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, *F15, *F16, *F17, *F18, *F19, *F20, ***F21**, *F22, *F23, *F24, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional
PF22	PF22 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, *F15, *F16, *F17, *F18, *F19, *F20, *F21, ***F22**, *F23, *F24, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional
PF23	PF23 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, *F15, *F16, *F17, *F18, *F19, *F20, *F21, *F22, ***F23**, *F24, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional
PF24	PF24 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, *F15, *F16, *F17, *F18, *F19, *F20, *F21, *F22, *F23, ***F24**, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional
PA1PF1	PA1-PF1 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, *F15, *F16, *F17, *F18, *F19, *F20, *F21, *F22, *F23, *F24, ***F1**, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional
PA1PF2	PA1-PF2 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, *F15, *F16, *F17, *F18, *F19, *F20, *F21, *F22, *F23, *F24, *F1, ***F2**, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional
PA1PF3	PA1-PF3 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, *F15, *F16, *F17, *F18, *F19, *F20, *F21, *F22, *F23, *F24, *F1, *F2, ***F3**, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional
PA1PF4	PA1-PF4 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, *F15, *F16, *F17, *F18, *F19, *F20, *F21, *F22, *F23, *F24, *F1, *F2, *F3, ***F4**, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional
PA1PF5	PA1-PF5 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, *F15, *F16, *F17, *F18, *F19, *F20, *F21, *F22, *F23, *F24, *F1, *F2, *F3, *F4, ***F5**, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional
PA1PF6	PA1-PF6 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, *F15, *F16, *F17, *F18, *F19, *F20, *F21, *F22, *F23, *F24, *F1, *F2, *F3, *F4, *F5, ***F6**, *F7, *F8, *F9, *F10, *F11, *F12	Optional
PA1PF7	PA1-PF7 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, *F15, *F16, *F17, *F18, *F19, *F20, *F21, *F22, *F23, *F24, *F1, *F2, *F3, *F4, *F5, *F6, ***F7**, *F8, *F9, *F10, *F11, *F12	Optional

Keyword	Description	Choices	Notes
PA1PF8	PA1-PF8 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, *F15, *F16, *F17, *F18, *F19, *F20, *F21, *F22, *F23, *F24, *F1, *F2, *F3, *F4, *F5, *F6, *F7, ***F8**, *F9, *F10, *F11, *F12	Optional
PA1PF9	PA1-PF9 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, *F15, *F16, *F17, *F18, *F19, *F20, *F21, *F22, *F23, *F24, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, ***F9**, *F10, *F11, *F12	Optional
PA1PF10	PA1-PF10 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, *F15, *F16, *F17, *F18, *F19, *F20, *F21, *F22, *F23, *F24, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, ***F10**, *F11, *F12	Optional
PA1PF11	PA1-PF11 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, *F15, *F16, *F17, *F18, *F19, *F20, *F21, *F22, *F23, *F24, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, ***F11**, *F12	Optional
PA1PF12	PA1-PF12 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, *F15, *F16, *F17, *F18, *F19, *F20, *F21, *F22, *F23, *F24, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, ***F12**	Optional
PA2PF1	PA2-PF1 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, ***F13**, *F14, *F15, *F16, *F17, *F18, *F19, *F20, *F21, *F22, *F23, *F24, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional
PA2PF2	PA2-PF2 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, ***F14**, *F15, *F16, *F17, *F18, *F19, *F20, *F21, *F22, *F23, *F24, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional
PA2PF3	PA2-PF3 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, ***F15**, *F16, *F17, *F18, *F19, *F20, *F21, *F22, *F23, *F24, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional
PA2PF4	PA2-PF4 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, *F15, ***F16**, *F17, *F18, *F19, *F20, *F21, *F22, *F23, *F24, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional
PA2PF5	PA2-PF5 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, *F15, *F16, ***F17**, *F18, *F19, *F20, *F21, *F22, *F23, *F24, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional
PA2PF6	PA2-PF6 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, *F15, *F16, *F17, ***F18**, *F19, *F20, *F21, *F22, *F23, *F24, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional
PA2PF7	PA2-PF7 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, *F15, *F16, *F17, *F18, ***F19**, *F20, *F21, *F22, *F23, *F24, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional

Keyword	Description	Choices	Notes
PA2PF8	PA2-PF8 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, *F15, *F16, *F17, *F18, *F19, ***F20**, *F21, *F22, *F23, *F24, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional
PA2PF9	PA2-PF9 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, *F15, *F16, *F17, *F18, *F19, *F20, ***F21**, *F22, *F23, *F24, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional
PA2PF10	PA2-PF10 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, *F15, *F16, *F17, *F18, *F19, *F20, *F21, ***F22**, *F23, *F24, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional
PA2PF11	PA2-PF11 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, *F15, *F16, *F17, *F18, *F19, *F20, *F21, *F22, ***F23**, *F24, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional
PA2PF12	PA2-PF12 key value	*ATTN, *HELP, *HLP3270, *CLEAR, *PRINT, *DSPATR, *TEST, *DOWN, *UP, *NONE, *RESET, *SYSREQ, *BCKSPC, *F13, *F14, *F15, *F16, *F17, *F18, *F19, *F20, *F21, *F22, *F23, ***F24**, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12	Optional

Top

Device (DEV)

Specifies a valid 3270 display station that is assigned this keyboard mapping function.

The possible values are:

***REQUESTER**

This mapping is assigned to the device where the command is entered.

device-name

Specify the name of the device to which the new keyboard mapping is to apply. You must have allocation rights to the specified device.

Top

PF1 key value (PF1)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**

5250 Help

***HLP3270**

3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PF2 key value (PF2)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PF3 key value (PF3)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PF4 key value (PF4)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PF5 key value (PF5)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PF6 key value (PF6)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PF7 key value (PF7)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PF8 key value (PF8)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PF9 key value (PF9)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PF10 key value (PF10)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PF11 key value (PF11)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PF12 key value (PF12)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PF13 key value (PF13)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PF14 key value (PF14)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PF15 key value (PF15)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PF16 key value (PF16)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PF17 key value (PF17)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PF18 key value (PF18)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PF19 key value (PF19)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PF20 key value (PF20)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PF21 key value (PF21)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PF22 key value (PF22)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PF23 key value (PF23)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PF24 key value (PF24)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PA1-PF1 key value (PA1PF1)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PA1-PF2 key value (PA1PF2)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PA1-PF3 key value (PA1PF3)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PA1-PF4 key value (PA1PF4)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PA1-PF5 key value (PA1PF5)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PA1-PF6 key value (PA1PF6)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PA1-PF7 key value (PA1PF7)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PA1-PF8 key value (PA1PF8)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PA1-PF9 key value (PA1PF9)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PA1-PF10 key value (PA1PF10)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PA1-PF11 key value (PA1PF11)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PA1-PF12 key value (PA1PF12)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PA2-PF1 key value (PA2PF1)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PA2-PF2 key value (PA2PF2)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PA2-PF3 key value (PA2PF3)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PA2-PF4 key value (PA2PF4)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:
- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PA2-PF5 key value (PA2PF5)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PA2-PF6 key value (PA2PF6)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PA2-PF7 key value (PA2PF7)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PA2-PF8 key value (PA2PF8)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PA2-PF9 key value (PA2PF9)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PA2-PF10 key value (PA2PF10)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PA2-PF11 key value (PA2PF11)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

PA2-PF12 key value (PA2PF12)

Each valid key or key sequence is a separate parameter which, except for restrictions as noted, can be assigned any function. If you do not specify a function for a particular key or key sequence, the function currently assigned to that key or key sequence remains the same.

The possible functions that can be assigned are:

***HELP**
5250 Help

***HLP3270**
3270 Help Text (Display Active Keyboard Map)

***CLEAR**
Clear Screen

***PRINT**
Print Screen

***DSPATR**
Display Imbedded Attributes

***TEST** Test Request

***DOWN**
Roll Down

***UP** Roll Up

***NONE**
No Assignment

***RESET**
Error Reset

***SYSREQ**
System Request

***BCKSPC**
Record Backspace

***ATTN**
Attention

***F1-*F24**
F1 through F24 Function Keys

Restrictions:

*HELP, *HLP3270, and *RESET must be assigned to one of the following 12-key sets of function keys:

- PF1 through PF12
- PA1/PF1 through PF12
- PA2/PF1 through PF12

The *HELP, *HLP3270, and *RESET functions are required for 3270 display station device support to function properly. Since not all 3270 keyboards have 24 PF keys, *HELP, *HLP3270, or *RESET may not be assigned to keys PF13 through PF24, unless these functions are **also** assigned to one of the three sets listed above.

It is recommended that *F1 and *SYSREQ both also be assigned to one of the three sets.

The value *ATTN cannot be explicitly assigned to a 3270 remote attach display station. If the default value *ATTN is taken, the value *NONE is substituted. However, if the value *ATTN is explicitly chosen, a diagnostic message is sent.

Top

Examples

```
SETKBDMAP   PF1(*F1)  PF2(*F2)  PF3(*F3)
            PF4(*F4)  PF5(*HLP3270)  PF9(*HELP)
```

This command reassigns the keyboard primarily for an application that makes frequent use of the 5250 CF keys F1, F2, F3, F4. All other PF key sequences are set to the default shown on the command prompt. The above command is started in the program that started the application (thus tailoring the display station to whatever application is run).

Top

Error messages

***ESCAPE Messages**

CPF8701
Specified device &1 not a 3270 device type.

CPF8702
&1 function key not correctly defined.

CPF8703
Device &1 not ready.

Top

Set Master Key (SETMSTKEY)

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

Parameters
Examples
Error messages

The Set Master Key (SETMSTKEY) command sets the specified master key from the parts already added. Master key parts can be added with the Add Master Key Part (ADDMSTPART) CL command, the Qc3LoadMasterKeyPart API, or Manage Master Keys panel in System i Navigator. Upon successful completion of this command, any keys encrypted under this master key should be retranslated. In order to save the master keys, a Save System (SAVSYS) command must be run.

For more information on master keys, refer to the Cryptographic Services Master Keys article in the Cryptographic Services section of the APIs topic collection in the Programming category in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/.

Restrictions:

- You must have all object (*ALLOBJ) and security administrator (*SECADM) special authorities to run this command.

Top

Parameters

Keyword	Description	Choices	Notes
MSTKEY	Master key	1-8, *ASP, *SAVRST	Required, Positional 1

Top

Master key (MSTKEY)

Specifies the master key on which to perform the action.

This is a required parameter.

The action will be performed on:

***ASP** The master key used for encrypting data stored on auxiliary storage pool (ASP) disk storage.

***SAVRST**
The master key used for encrypting all the other master keys on a SAVSYS operation.

1-8 One of the eight general purpose master keys.

Top

Examples

```
SETMSTKEY   MSTKEY(3)
```

This command first moves the current version of Master Key 3 into the old version, and then moves the new version (consisting of all parts added for Master Key 3 since the last Set Master Key) into the current version.

Top

Error messages

***ESCAPE Messages**

CPF222E
&1 special authority is required.

CPF3CF2
Error(s) occurred during running of &1 API.

CPF9872
Program or service program &1 in library &2 ended. Reason code &3.

CPF9D88
An error occurred during exit program post-processing.

CPF9D89
An error occurred during exit program pre-processing.

CPF9D90
Master Key &1 was not set due to an exit program cancel.

CPF9D94
A pending value exists for a master key.

CPF9DB0
No key parts have been loaded.

CPF9DDA
Unexpected return code &1 from cryptographic service provider &2.

Top

Set Object Access (SETOBJACC)

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

Parameters
Examples
Error messages

The Set Object Access (SETOBJACC) command temporarily changes the speed of access to an object by bringing the object into a main storage pool or purging it from all main storage pools. An object can be kept main storage resident by selecting a pool for the object that has available space and does not have jobs associated with it. Repeated use of the command can cause a set of objects to be resident in a main storage pool.

Restrictions:

- You must have object operational (*OBJOPR) authority to all objects that are brought into a main storage pool or purged from all main storage pools.

Top

Parameters

Keyword	Description	Choices	Notes
OBJ	Object	*Qualified object name*	Required, Positional 1
	Qualifier 1: Object	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *USRLIBL, *CURLIB, *ALL, *ALLUSR	
OBJTYPE	Object type	*FILE, *PGM	Optional, Positional 2
POOL	Storage pool	*Element list*	Optional, Positional 3
	Element 1: Shared pool or subsystem name	*Name*, *JOB, *BASE, *SHRPOOL1, *SHRPOOL2, *SHRPOOL3, *SHRPOOL4, *SHRPOOL5, *SHRPOOL6, *SHRPOOL7, *SHRPOOL8, *SHRPOOL9, *SHRPOOL10, *SHRPOOL11, *SHRPOOL12, *SHRPOOL13, *SHRPOOL14, *SHRPOOL15, *SHRPOOL16, *SHRPOOL17, *SHRPOOL18, *SHRPOOL19, *SHRPOOL20, *SHRPOOL21, *SHRPOOL22, *SHRPOOL23, *SHRPOOL24, *SHRPOOL25, *SHRPOOL26, *SHRPOOL27, *SHRPOOL28, *SHRPOOL29, *SHRPOOL30, *SHRPOOL31, *SHRPOOL32, *SHRPOOL33, *SHRPOOL34, *SHRPOOL35, *SHRPOOL36, *SHRPOOL37, *SHRPOOL38, *SHRPOOL39, *SHRPOOL40, *SHRPOOL41, *SHRPOOL42, *SHRPOOL43, *SHRPOOL44, *SHRPOOL45, *SHRPOOL46, *SHRPOOL47, *SHRPOOL48, *SHRPOOL49, *SHRPOOL50, *SHRPOOL51, *SHRPOOL52, *SHRPOOL53, *SHRPOOL54, *SHRPOOL55, *SHRPOOL56, *SHRPOOL57, *SHRPOOL58, *SHRPOOL59, *SHRPOOL60, *PURGE	
	Element 2: Pool identifier	1-10	
MBR	Member	*Name*, ***FIRST**	Optional
MBRDATA	Member data	***BOTH**, *ACCPTH, *DATA	Optional

Top

Object (OBJ)

Specifies the qualified name of the object to be brought into or deleted from main storage.

The name of the specified object 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.

***USRLIBL**
Only the libraries in the user portion of the job's library list are searched.

***ALL** All libraries on the system 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
#COBLIB
#DFULIB
#DSULIB
#RPGLIB
#SDALIB
#SEULIB

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
QGPL
QGPL38
QMGTC
QMGTC2
QMPGDATA
QMQMDATA
QMQMPROC
QPFRDATA
QRCL
QRCLxxxxx
QSRVAGT
QSYS2
QSYS2xxxxx
QS36F
QUSER38
QUSRADSM
QUSRBRM
QUSRDIRCF
QUSRDIRCL
QUSRDIRDB
QUSRIJS
QUSRINFSKR
QUSRNOTES
QUSROND
QUSRPOSGS
QUSRPOSSA
QUSRPYMSVR
QUSRRDARS
QUSRSYS
QUSRVI
QUSRVxRxMx

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.

The possible values are:

object-name
Specify the name of the object.

Top

Object type (OBJTYPE)

Specifies the type of object to be brought into or deleted from main storage.

The possible values are:

***FILE** The object is a file.

***PGM** The object is a program.

Top

Storage pool (POOL)

Specifies whether the object is brought into or purged from main storage.

The possible values are:

***JOB** The object is brought into the pool associated with the job.

***BASE**
The object is brought into the base pool.

***SHRPOOLn**
The object is brought into a a general-purpose shared pool. Valid values range from 1 through 10.

***PURGE**
The object is purged from all pools. **Element 1: Subsystem**

subsystem
Specify a subsystem name. **Element 2: Pool Identifier**

pool-identifier
Specify a subsystem pool identifier.

***PURGE**
The object is purged from all pools.

Top

Member (MBR)

Specifies the database file member to be brought into or purged from main storage.

The possible values are:

***FIRST**
The first member is selected.

file-member-name
Specify the member name.

Top

Member data (MBRDATA)

Specifies the member data to be brought into or purged from main storage.

The possible values are:

***BOTH**
All parts of the object are selected.

***ACCPTH**
The file member's access path is selected.

***DATA**
The file member's data is selected.

Top

Examples

```
SETOBJACC   OBJ(OBJA)  OBJTYPE(*PGM)  POOL(*JOB)
```

This command brings a program named OBJA to the pool associated with the job in which the command is run.

Top

Error messages

***ESCAPE Messages**

CPF1858
The specified pool does not exist.

CPF1859
Use of an access path was requested but none exists.

CPF9855
File &1 in library &3 contains no members.

Top

Set Program Information (SETPGMINF)

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

Parameters
Examples
Error messages

The Set Program Information (SETPGMINF) command is used with the extended program model (EPM) languages to associate all the program objects in an application. The SETPGMINF command defines the application environment, based on the information you supply on the ROOTPGM, SUBPGM, and LIBFILE parameters. The C/400*, FORTRAN/400*, and Pascal languages work within the extended program model.

EPM programs that refer to external symbols in other EPM programs must be specified on the SETPGMINF command. External symbols are calls to other compilation units or external storage. You do not need to use this command if your program consists of only one compilation unit, or if the program calls only non-EPM programs. If the programs you specify on the ROOTPGM and SUBPGM parameters have not been successfully compiled, SETPGMINF fails.

Error messages for SETPGMINF

***ESCAPE Messages**

PSE4017
Errors occurred in SETPGMINF command.

Top

Parameters

Keyword	Description	Choices	Notes
ROOTPGM	Root program	*Qualified object name*	Required, Positional 1
	Qualifier 1: Root program	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
SUBPGM	Sub-programs	Values (up to 200 repetitions): *Element list*	Optional, Positional 2
	Element 1: Program	*Qualified object name*	
	Qualifier 1: Program	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
LIBFILE	Library information files	Values (up to 16 repetitions): *Element list*	Optional, Positional 3
	Element 1: File	*Qualified object name*	
	Qualifier 1: File	*Name*, ***SAME**, *NONE, *PASLIB, *FTNLIB, *CLIB	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
LISTDETAIL	Listing detail	***NONE**, *BASIC, *FULL	Optional
PRTFILE	Print file	*Qualified object name*	Optional
	Qualifier 1: Print file	*Name*, **QSYSPRT**	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	

Keyword	Description	Choices	Notes
RUNATTR	Run attributes	*Element list*	Optional
	Element 1: Maximum non-fatal errors	0-100, **20**, *NOMAX	
	Element 2: Fatal error severity	0-40, **40**	
	Element 3: External Type Checking	***YES**, *NO	
	Element 4: Computational Attributes	***LANG**, *ALL, *NONE	
PFROPT	Performance options	*Element list*	Optional
	Element 1: Access group storage	***NONE**, *ALL	
HEAPSIZE	Initial size of heap spaces	*Element list*	Optional
	Element 1: Dynamic storage heap	1024-16777216, **16000**, *NONE	
	Element 2: Static storage heap	1024-16777216, **32000**	
STACKSIZE	Initial size of auto storage	*Element list*	Optional
	Element 1:	1024-16777216, **16000**	
SSNATTR	Session file attributes	*Element list*	Optional
	Element 1: File size	8192-16000000, **32000**	
	Element 2: Buffer size	80-1024, **160**	
DBGOPT	Debug options	*Element list*	Optional
	Element 1: Debug initialization	***ON**, *OFF	

Top

Root program (ROOTPGM)

Specifies the name of the program which will contain the environment definition necessary to create the EPM run-time environment. This program, called a default entry point or root program, contains the references to external symbols. In C, the default entry point is usually the program that contains a main() function. Pascal defines the main begin block of the program unit as the default entry point. FORTRAN defines the main program (the program defined on the PROGRAM statement) as the default entry point. *Change authority is required.

ROOTPGM is a required parameter.

program-name
The name of the EPM program object that contains the default entry point.

Possible library values are:

***LIBL** The system searches the library list for the root program object.

***CURLIB**
The current library is searched. If you have not specified the current library, QGPL is used.

library-name
Enter the name of the library where the root program object is located.

Top

Sub-programs (SUBPGM)

Specifies a list of all the program objects that you want to include in the run-time environment for your application. These programs can be either EPM or non-EPM program objects.

Up to 200 program objects can be specified on the SUBPGM parameter. You can specify more than 200 related program objects for 1 application if you nest SETPGMINF commands.

program-name
Enter the name of the program object. You can specify up to 200 program objects on one SETPGMINF command.

Possible library values are:

***LIBL** The system searches the library list for the library containing the program objects.

***CURLIB**
The name of the current library is used. If you have not specified the current library, QGPL is used.

library-name
Enter the name of the library where the program object is located.

These SUBPGMs are sought at run-time, according to the library specifications used when you specify the SETPGMINF command. If you specify a specific library, or use *CURLIB, only that specific library is searched for the program object. If the library designated as the current library changes between the time you issue the SETPGMINF command and the time you run your program, not all of the program objects will be found and you will receive an error message.

Top

Library information files (LIBFILE)

Specifies the library information file that contains information used to resolve any outstanding external references after all of the program objects identified on the ROOTPGM and SUBPGM parameters have been processed. Library information files are searched in the order that they are specified.

Several library information files are IBM-supplied. These files contain the EPM language library functions.

You can create and update your own library information file with the Extract Program Information (EXTPGMINF) command.

***SAME**
Use the same library information file as was used in the last environment definition. If this is the first time an environment is defined, the run-time library file for the language object specified on the ROOTPGM parameter is used. For example, if your ROOTPGM was compiled using the C/400 compiler, the default library information file is *CLIB.

***NONE**
No library information file is used.

***PASLIB**
The Pascal run-time library information file is used.

***CLIB** The C/400 run-time library information file is used.

***FTNLIB**
The FORTRAN/400 run-time library information file is used.

file-name
Enter the name of the library information file. This library information file must exist. To create a library information file, use the EXTPGMINF command.

The possible library values are:

***LIBL** The system searches the library list for the specified library information file.

***CURLIB**
The current library is used to locate the library information file. If you have not specified the current library, QGPL is used.

library-name
Enter the name of the library that contains the library information file.

Top

Listing detail (LISTDETAIL)

Specifies whether a SETPGMINF listing is created. The listing will have the same name as the program specified on the ROOTPGM parameter and is directed to the library and file specified on the PRTFILE parameter.

The PRTFILE parameter is not displayed unless you request a listing (*BASIC or *FULL) on the LISTDETAIL parameter.

***NONE**
A listing is not created.

***BASIC**
A listing is created that includes all of the symbol references and definitions that result from the programs specified on the ROOTPGM and SUBPGM parameters.

***FULL** A listing is created that includes all of the symbol references and definitions for the entire application. This includes the symbol references and definitions for all the programs specified on the LIBFILE parameter that are not explicitly referenced in your application.

Top

Print file (PRTFILE)

Specifies the name and library of the printer file where the SETPGMINF listing is directed. The file should have a minimum length of 132. If you specify a file with a record length of less than 132, information may be lost.

This parameter does not appear on the prompting display unless you change the default value on the LISTDETAIL parameter to *BASIC or *FULL.

QSYSPRT
The SETPGMINF listing is placed in the file QSYSPRT.

file-name
Enter the name of the file where the SETPGMINF listing is to be placed.

The possible library values are:

***LIBL** The system searches the library list.

***CURLIB**
The name of the current library is used. If you have not specified the current library, QGPL is used.

library-name
Enter the name of the library where the file is stored.

Top

Run attributes (RUNATTR)

Specifies the number of times the non-fatal error counter is incremented before processing ends, and the message severity-level that is interpreted as a fatal error at run-time. A fatal error is an error that stops your application from running. You can also specify whether you want external type checking to take place at program run-time. The computational attributes field allows you to set the attributes for controlling floating point operations in the run-time environment.

20 The non-fatal error counter is incremented up to 20 times before the processing of your application ends.

***NOMAX**
The non-fatal error counter has no limit and will not stop the processing of your application.

counter-number
The number of non-fatal errors that are allowed before processing ends.

40 A message with a severity-level of 40 or higher is interpreted as a fatal error.

severity-level
The message severity-level that is interpreted as a fatal error. If a severity-level of 0 is specified, your application stops running if any errors occur.

***YES** External type checking is performed at program run-time.

***NO** External type checking is not performed at program run-time. Any warning messages that occur as a result of checking errors are not issued when you run your program.

***LANG**
Computational attributes are set according to the semantics of the EPM language.

***ALL** Floating point exceptions are disabled or masked.

***NONE**
Floating point exceptions are enabled or unmasked.

Top

Performance options (PFROPT)

Specifies whether the space allocated for static, automatic, and dynamic program variables is part of the process access group.

***NONE**
The storage spaces created are not part of the process access group (PAG).

***ALL** The storage spaces created are part of the process access group (PAG). This option can improve the performance of programs that use limited program variable storage, and are run on a non-dedicated system. Information on performance tools and CL commands you can use to display and analyze the PAG can be found in the Performance Tools Guide.

Top

Initial size of heap spaces (HEAPSIZE)

Specifies the initial size of the heap spaces for the dynamic and static storage heap. The heap size specified will expand as your application progresses.

The dynamic storage heap consists of the storage that you have allocated in your application with the C/400 memory routines (malloc, calloc, realloc) or the NEW procedure in Pascal.

The static storage heap consists of the storage that you have allocated in your application with static and external static variables.

16000 The initial size of the dynamic storage heap is 16000 bytes.

***NONE**
No initial dynamic storage is allocated. Use *NONE only if you do not use any EPM memory management functions in your application. If you specify *NONE and use any of the memory functions, you will get pointer exceptions and your application will stop processing.

dynamic-storage-size
Enter the initial size of the dynamic storage heap. The initial size can be between 1024 and 16777216 bytes.

32000 The initial size of the static storage heap is 32000 bytes.

static-storage-size
Enter the initial size of the static storage heap. The initial size can be between 1024 and 16777216 bytes. If you use static variables extensively in your application your performance can improve if you specify a larger storage heap than the default.

Top

Initial size of auto storage (STACKSIZE)

Specifies the initial size of the automatic storage stack. The stack size specified will expand as your application progresses.

The automatic storage stack consists of the storage that you have allocated in your application with automatic variables.

16000 The initial size of the automatic storage stack is 16000 bytes.

automatic-storage-size
Enter the initial size of the automatic storage stack. The initial size can be between 1024 and 16777216 bytes. If your application contains a large number of recursive calls, you should increase the initial size of the stack.

Top

Session file attributes (SSNATTR)

Specifies the file session attributes for the file and buffer size.

32000 The default size for the file is 32000 bytes.

file-size
Specify a value for the file size between 8192 and 16000000 bytes. If your file exceeds the size you specify, the information at the top rolls off the display. This information is lost.

160 The default size for the buffer is 160 bytes.

buffer-size
Specify a value for the buffer size between 80 and 1024 bytes.

Top

Debug options (DBGOPT)

Specifies the use of the extended program model (EPM) debug tool at run-time, if system debug mode is active. To activate system debug mode, enter the STRDBG command.

***ON** EPM debug starts at run-time if system debug mode is active.

***OFF** EPM debug will not start at run-time. You can use system debug to debug your programs, but EPM language variable names, statement numbers, and debugging commands are not available.

Top

Examples

None

Top

Error messages

***ESCAPE Messages**

PSE4017
Errors occurred in SETPGMINF command.

Top

Set Tape Category (SETTAPCGY)

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

Parameters
Examples
Error messages

The Set Tape Category (SETTAPCGY) command sets the category for a tape device in a specified media library device. The system automatically loads cartridges from the specified category in the order specified on the cartridge order (CTGORDER) parameter.

Top

Parameters

Keyword	Description	Choices	Notes
DEV	Library device	*Name*	Required, Positional 1
OPTION	Option	*MOUNTED, *DEMOUNTED, *ASSIGN, *RELEASE	Required, Positional 2
CGY	Category	Single values: *SHARE400 Other values: *Element list*	Optional
	Element 1: Category name	*Character value*, ***NOSHARE**, *IPL, *NL, *CNV	
	Element 2: Category system	*Character value*, ***CURRENT**	
CTGORDER	Cartridge order	***SEQ**, *NEXTAVAIL	Optional
TGTCGY	Target category	Single values: *SHARE400 Other values: *Element list*	Optional
	Element 1: Category name	*Character value*, ***CGY**, *NOSHARE, *IPL, *NL	
	Element 2: Category system	*Character value*, ***CURRENT**	
MNTID	Mount identifier	*Name*, ***NONE**	Optional

Top

Library device (DEV)

Specifies the media library device for which the category is set.

This is a required parameter.

name Specify the name of the media library device.

Top

Option (OPTION)

Specifies which option is set for the specified media library device.

This is a required parameter.

***MOUNTED**
The category specified on the CGY parameter is considered mounted and any tape operation with VOL(*MOUNTED) specified uses the tape cartridges from the category that is mounted.

***DEMOUNTED**
The category specified on the CGY parameter is no longer mounted. The use of VOL(*MOUNTED) is not valid for the media library device.

***ASSIGN**
The mounted category session specified on the MNTID parameter is assigned to the job issuing the SETTAPCGY command. The mounted category session being assigned must have been previously mounted and released.

***RELEASE**
The mounted category session assigned to the job issuing the SETTAPCGY command is released and is available for another job to assign.

Top

Category (CGY)

Specifies the category to be mounted.

Single values

***SHARE400**
The cartridge identifier can be shared by other systems that are attached to the library device. The cartridge identifiers are mounted in the order specified in the CTGORDER parameter.

Element 1: Category name

***NOSHARE**
The cartridge identifiers cannot be shared with other systems that are attached to the same device. The cartridge identifiers are mounted in the order specified in the CTGORDER parameter.

***IPL** The cartridge identifiers can be used for an alternate initial program load (IPL) of the system. The cartridge identifiers are mounted in the order specified in the CTGORDER parameter.

***NL** The cartridge is used as a non-labeled tape. The cartridge identifiers are mounted in the order specified in the CTGORDER parameter.

***CNV** The cartridge identifier is used from the special convenience category. The cartridge identifiers are mounted in the order specified in the CTGORDER parameter.

character-value
Specify the name of a user-defined category. The cartridge identifiers in the category specified are mounted in the order specified in the CTGORDER parameter.

Element 2: Category system

The second part of the parameter specifies the name of the system to which the category belongs. The system name is obtained from the pending system name field of a Display Network Attributes (DSPNETA) command. If there is no pending system name, the current system name attribute is used.

```
* * * Attention * * *
If a system name is changed, the tape cartridges in
library devices that have the attribute of the system
name before it was changed are no longer valid.
```

***CURRENT**
The category belongs to the system currently running the command.

character-value
Specify the name of the system to which the category belongs.

Top

Cartridge order (CTGORDER)

Specifies the order in which the cartridges are mounted.

***SEQ** The cartridges are mounted in the order they were added or changed to the category specified. The operation ends if the next cartridge in the sequential order is not available.

***NEXTAVAIL**
The cartridges are mounted in a sequential order, but if a cartridge in the order is not available, the next available cartridge is used.

Top

Target category (TGTCGY)

Specifies the category to which a tape cartridge is changed after it is used. This parameter can be useful when a scratch category is set for use during a save operation. Each cartridge is automatically changed to the specified target category after it is used.

Single values

***SHARE400**
The cartridge identifiers are changed to the *SHARE400 category.

Element 1: Category name

***CGY** The cartridges remain in the category specified on the CGY parameter.

***NOSHARE**
The cartridge identifiers are changed to the *NOSHARE category.

***IPL** The cartridge identifiers are changed to the *IPL category.

***NL** The cartridge identifiers are changed to the *NL category.

character-value
Specify the name of a user-defined category. The cartridge identifiers are changed to the specified user-defined category.

Element 2: Category system

The second part of this parameter specifies the name of the system to which the target category belongs. The system name is obtained from the pending system name field of the Display Network Attributes (DSPNETA) command. If there is no pending system name, the current system name attribute is used.

```
* * * Attention * * *
If a system name is changed, the tape cartridges in
library devices that have the attribute of the system
name before it was changed are no longer valid.
```

***CURRENT**
The system currently running the command is used.

character-value
Specify the name of the system that the category belongs to.

Top

Mount identifier (MNTID)

Specifies the identifier by which the mounted category session will be known. This parameter is valid only when OPTION(*MOUNTED) or OPTION(*ASSIGN) is specified.

***NONE**
This mounted category session is not assigned to any job and will be used by the first job that issues a command to the media library with a volume identifier of *MOUNTED. The mount identifier *NONE cannot be specified with OPTION(*ASSIGN).

name Specify a unique name to identify the mounted category session. A mounted category session known by this mount identifier is created and assigned when the category is mounted. This identifier is then used when assigning the session to another job. The session known by this identifier is deleted when the category is demounted. The mount identifier can then be reused when mounting another category to the media library.

Top

Examples

Example 1: Using Mounted Category Without a Mount Identifier

```
SETTAPCGY   MLB(LIB01)  OPTION(*MOUNTED)  CGY(*NOSHARE *CURRENT)
            CTGORDER(*SEQ)  TGTCGY(*IPL)
```

This command sets the tape category to *NOSHARE for a resource in media library device LIB01 on the system currently running this command. Each cartridge that is used is changed to the *IPL category. The order in which the cartridges are used is the exact order in which they were added to or changed in the category. As each cartridge is used and unloaded from the resource, the system automatically chooses and loads the next sequential cartridge from the *NOSHARE category.

Example 2: Using Mounted Category with a Mount Identifer

```
SETTAPCGY DEV(LIB01) OPTION(*MOUNTED) CGY(*NOSHARE *CURRENT)
          CTGORDER(*SEQ) TGTCGY(DAILY1) MNTID(DAILY)
```

This command sets the tape category to *NOSHARE for a resource in media library device LIB01 on the system currently running this command. The mounted category session is identified by the mount identifier DAILY. Each cartridge that is used is changed to the DAILY1 category. The order in which the cartridges are used is the exact order in which they were added to or changed in the category. As each cartridge is used and unloaded from the resource, the system automatically chooses and loads the next sequential cartridge from the *NOSHARE category.

Example 3: Releasing a Category Session

```
SETTAPCGY   MLB(LIB01)  OPTION(*RELEASE)
```

This command releases the category session assigned to the job issuing the command. The category is still set to a resource in media library device LIB01 and is available for another job to assign.

Example 4: Assigning a Category Session

```
SETTAPCGY   MLB(LIB01)  OPTION(*ASSIGN)  MNTID(DAILY)
```

This command assigns the mounted category session identified by the mount identifier DAILY to the job issuing the command.

Example 5: Demounting a Mounted Category

```
SETTAPCGY   MLB(LIB01)  OPTION(*DEMOUNTED)
```

This command demounts the mounted category from a resource in media library device LIB01. The mount identifier DAILY is now available to use to name another mounted category session.

Top

Error messages

***ESCAPE Messages**

CPF6711
Command not allowed

CPF6712
Category &4 not mounted.

CPF6713
Category not demounted.

CPF6745
Device &1 not a media library device.

CPF67A6
Category does not exist

CPF67AD
Category not assigned.

CPF67AE
Category not released.

CPF67E4
Library device function not successful

CPF9814
Device &1 not found.

CPF9825
Not authorized to device &1.

Top

Set Upgrade Environment (SETUPGENV)

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

Parameters
Examples
Error messages

The Set Upgrade Environment (SETUPGENV) command prompts the user for information that is required to help the user plan for and perform the upgrade. The information that is gathered is stored in library QUPGRADE.

Top

Parameters

Keyword	Description	Choices	Notes
UPGENV	Upgrade environment	*Character value*, *NEW, *UPGENVID	Required, Positional 1
TGTRLS	Target release	*Character value*	Optional, Positional 2
TGTPRC	Target processor	*Character value*, ***DFT**	Optional, Positional 3
MERGE	Merge with Advanced/36	***NO**, *YES	Optional, Positional 4
UPGENVID	Upgrade environment identifier	*Name*, *SRLNBR	Optional, Positional 5

Top

Upgrade environment (UPGENV)

Specifies whether a new upgrade environment is being created or if you want to work with an existing environment.

***NEW** Specifies a new upgrade environment is being created.

***UPGENVID**
Specifies that you want to work with an existing upgrade environment. The identifier of the existing upgrade environment is specified in the upgrade environment identifier field.

Top

Target release (TGTRLS)

Specifies the target release to which you intend to upgrade. This field is only valid when the Upgrade Environment (UPGENV) value is *NEW.

target-release
Specify the release level in the format VxRxMx. Valid values depend on the current version, release, and modification level, and they change with each new release.

Top

Target processor (TGTPRC)

Specify the feature code of the iSeries processor you expect for your target system. This field is only valid when the Upgrade Environment (UPGENV) value is *NEW.

The processors listed are the valid target processors for upgrading your source iSeries system. Other valid processors may be entered but the upgrade would represent an unsupported upgrade.

***DFT** Specifies the smallest processor necessary to provide similar performance to that of the source system.

target-processor
Specifies the four digit feature code of the target processor. Valid values depend on the source and target releases.

Top

Merge with Advanced/36 (MERGE)

Indicates whether System/36 operating system (SSP) will be running as guest on the target system. This field is only valid when the Upgrade Environment (UPGENV) value is *NEW

***NO** Indicates this system will not be merged with a System/36

***YES** Indicates this system will be merged with a System/36

Top

Upgrade environment identifier (UPGENVID)

Specifies the identifier associated with an upgrade environment. You may develop several upgrade environments as you are planning the order for your upgrade. Each environment may use different values and upgrade methods to help you decide which upgrade method and approach is best for you and your business. Only one upgrade environment will be used for upgrade preparation and the upgrade.

***SRLNBR**
Specifies the default name for the upgrade environment. The upgrade environment identifier for this environment will be the serial number of this system appended to the letter ’Q’, for example, Q0100A12.

upgrade-environment-identifier
Specifies a character name as the upgrade environment identifier.

Top

Examples

None

Top

Error messages

Unknown

Top

Sign Off (SIGNOFF)

Where allowed to run: Interactive environments (*INTERACT *IPGM *IREXX *EXEC)
Threadsafe: No

Parameters
Examples
Error messages

The Sign Off (SIGNOFF) command ends an interactive job or causes all jobs in a group to end. You enter this command to sign off at a work station.

Restrictions:

1. This command is valid only in an interactive job.
2. If the SIGNOFF command is issued in a CL program, all subsequent commands in the CL program are bypassed.

Top

Parameters

Keyword	Description	Choices	Notes
LOG	Job log	***NOLIST**, *LIST	Optional, Positional 1
DROP	Drop line	***DEVD**, *YES, *NO	Optional, Positional 2
ENDCNN	End connection	***NO**, *YES	Optional, Positional 3

Top

Job log (LOG)

Specifies whether the job log for this interactive job is deleted or is included in the job's spooled output for printing. This entry takes precedence over the log value specified for the job itself.

***NOLIST**
The information in the job log, which has already been displayed throughout the job, is no longer needed and is being deleted.

***LIST** The job log is spooled for printing, along with the job's remaining spooled output, if any.

Top

Drop line (DROP)

Specifies, for switched lines only, whether the switched line attached to the work station is disconnected (dropped) if no other work stations on the same line are signed on. This parameter is ignored if the work station is attached to a nonswitched line.

***DEVD**
The value specified for the **Drop line (DROP)** parameter of the work station's device description is assumed.

*YES The switched line is disconnected when the job is ended if no other work stations on the line are signed on.

*NO The switched line is not disconnected when the job is ended.

Top

End connection (ENDCNN)

Specifies whether to end the connection to the current system. Ending the connection allows the user to bypass the sign-on display of the target system and return to the source system. For communication functions that do not support this option, this parameter is ignored.

*NO The connection does not end. The sign-on display of the target system is shown.

*YES The connection ends and the user is returned to the source system. No sign-on screen or error messages are shown from the target system.

Top

Examples

Example 1: Signing Off and Ending an Interactive Job

```
SIGNOFF
```

This command signs off the user of the work station and ends the interactive job. The switched line is dropped only if specified in the device description of this work station and if no other work station on this line is active. An end-of-job message that gives the job start and stop times is written in the job's log.

Example 2: Printing the Job Log

```
SIGNOFF   LOG(*LIST)  DROP(*NO)
```

This command ends the interactive job, but the switched line is not released. The job log is printed with the job's spooled output.

Example 3: Signing Off and Ending the Connection

```
SIGNOFF   ENDCNN(*YES)
```

This command ends the connection and transfers the user back to the source system.

Top

Error messages

None

Top

Select Command (SLTCMD)

Where allowed to run: Interactive environments (*INTERACT *IPGM *IREXX *EXEC)
Threadsafe: No

Parameters
Examples
Error messages

The Select Command (SLTCMD) command allows you to display a list of commands from one or more libraries. From the Select Commands display, you can select a command that you want to prompt.

Restrictions:

- Only the libraries to which you have use (*USE) authority will be searched.
- Only the commands to which you have some authority will be shown on the display.
- To prompt a listed command, you must have *USE authority to the command.

Top

Parameters

Keyword	Description	Choices	Notes
CMD	Command	*Qualified object name*	Required, Positional 1
	Qualifier 1: Command	*Generic name, name*, *ALL	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB, *USRLIBL, *ALLUSR, *ALL	

Top

Command (CMD)

Specifies the commands to be shown on the Select Command display.

This is a required parameter.

Qualifier 1: Message queue

***ALL** All commands are shown.

generic-name
Specify the generic name of the commands to be shown. A generic name is a character string that contains one or more characters followed by an asterisk (*). If a generic name is specified, all commands that have names with the same prefix as the generic name are shown.

name Specify the name of the command to be shown.

Qualifier 2: Library

***LIBL** All libraries in the library list for the current thread are searched. All objects in these libraries with the specified object name are shown.

***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:

```
QDSNX       QRCLxxxxx   QUSRDIRDB   QUSRVI
QGPL        QSRVAGT     QUSRIJS     QUSRVxRxMx
QGPL38      QSYS2       QUSRINFSKR
QMGTC       QSYS2xxxxx  QUSRNOTES
QMGTC2      QS36F       QUSROND
QMPGDATA    QUSER38     QUSRPOSGS
QMQMDATA    QUSRADSM    QUSRPOSSA
QMQMPROC    QUSRBRM     QUSRPYMSVR
QPFRDATA    QUSRDIRCF   QUSRRDARS
QRCL        QUSRDIRCL   QUSRSYS
```

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.

Top

Examples

```
SLTCMD   CMD(QSYS/*ALL)
```

This command shows a list of all commands in library QSYS. The option to prompt and run commands is available.

Top

Error messages

None

Top

Send Break Message (SNDBRKMSG)

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

Parameters
Examples
Error messages

The Send Break Message (SNDBRKMSG) command is used to send an immediate message to one or more work station message queues. An immediate message is a message that is not predefined and is not stored in a message file. The command causes the message to be delivered always in break mode. The DSPMSG display is shown for the message when it is received, regardless of the setting of the message queue's delivery mode, severity, and break handling program. However, the message may not be displayed in some cases, depending on the BRKMSG job attribute. This command is primarily intended for the system operator's use.

Restrictions:

1. This command can be used to send break messages to work station message queues only.
2. This command cannot send inquiry messages (specified by MSGTYPE(INQ)) to multiple work stations.

Top

Parameters

Keyword	Description	Choices	Notes
MSG	Message text	*Character value*	Required, Positional 1
TOMSGQ	To work station message queue	Single values: *ALLWS Other values (up to 50 repetitions): *Qualified object name*	Required, Positional 2
	Qualifier 1: To work station message queue	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**	
MSGTYPE	Message type	***INFO**, *INQ	Optional, Positional 3
RPYMSGQ	Message queue to get reply	*Qualified object name*	Optional
	Qualifier 1: Message queue to get reply	*Name*, **QSYSOPR**	
	Qualifier 2: Library	*Name*, ***LIBL**	
CCSID	Coded character set ID	1-65535, *HEX, ***JOB**	Optional

Top

Message text (MSG)

Specifies the immediate message that is being sent. The text must be enclosed in apostrophes if it contains blanks or other special characters. A maximum of 512 characters can be specified.

Coded Character Set Identifier (CCSID) Considerations

The text supplied for the MSG parameter is assumed to be in the CCSID of the job running this command unless a coded character set identifier is supplied for the CCSID parameter. For more information about the message handler and its use of CCSIDs, see the i5/OS globalization topic collection in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/.

This is a required parameter.

Top

To work station message queue (TOMSGQ)

Specifies one or more work station message queues to which the break message is sent. Only the names of work station message queues can be specified and only *LIBL or QSYS can be specified for the library value.

Single values

***ALLWS**
The break message is sent to all work station and Personal Computer message queues. *ALLWS cannot be specified if *INQ is specified for the **Message type (MSGTYPE)** parameter.

Qualifier 1: To work station message queue

message-queue-name
Specify the name of the message queue to which the break message is to be sent.

Qualifier 2: Library

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

library-name
Specify the library where the message queue is located.

Top

Message type (MSGTYPE)

Specifies the type of message that is sent in break mode. Only informational or inquiry message types can be specified. Inquiry messages may require a response.

***INFO**
An information only message is sent in break mode.

***INQ** An inquiry message is sent in break mode; the work station receiving the message is expected to reply to it. An inquiry message cannot be sent to multiple queues with only one command.

Top

Message queue to get reply (RPYMSGQ)

Specifies, only if an inquiry message is sent, the message queue that the work station user’s reply is sent to.

Qualifier 1: Message queue to get reply

QSYSOPR
The replies to the break message are sent to the system operator’s message queue, QSYSOPR.

message-queue-name
Specify the name of the message queue to which a reply to the break message is sent. Only a user or work station message queue can be specified.

Qualifier 2: Library

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

library-name
Specify the library where the message queue is located.

Top

Coded character set ID (CCSID)

Specifies the coded character set identifier (CCSID) that the specified message text is in. The text supplied by the MSG parameter is assumed to be in the CCSID supplied by this parameter. For more information about the message handler and its use of CCSIDs, see the i5/OS globalization topic collection in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/.

***JOB** The message text is assumed to be in the CCSID of the job running this command.

***HEX** The message text is not converted. CCSID 65535 is used.

coded-character-set-identifier
Specify a valid CCSID in which you want your message text to be considered in. 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. This command will validate the CCSID.

Top

Examples

Example 1: Sending a Message

```
SNDBRKMSG   MSG('Inventory application shuts down at 4:00 PM.')
```

This command sends the message ’The inventory application shuts down at 4:00 pm today.’ to all work station message queues. If the work station is signed on, the message will be delivered in break mode regardless of the delivery attribute setting of those message queues. The message is also added to the work station message queues of those work stations that are not signed on.

Example 2: Sending an Immediate Message

```
SNDBRKMSG   MSG('Your printed output is ready.')
            TOMSGQ(GEORGEMSGQ)
```

This example shows a typical use of the SNDBRKMSG command by the system operator to send an immediate message to a work station user.

Top

Error messages

***ESCAPE Messages**

CPF2428
Message queue parameter is not valid.

CPF2469
Error occurred when sending message&1.

CPF247E
CCSID &1 is not valid.

CPF9838
User profile storage limit exceeded.

Top

Send Distribution (SNDDST)

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

Parameters
Examples
Error messages

The Send Distribution (SNDDST) command allows you to send a distribution to a user, to a list of users, or to a distribution list.

Restrictions:

- If you are working on behalf of another user, you must have been granted permission to work for that user through the Grant User Permission (GRTUSRPMN) command.
- You need to be enrolled in the system distribution directory.
- You cannot request personal distribution if you are working for another user.

Top

Parameters

Keyword	Description	Choices	Notes
TYPE	Information to be sent	*MSG, *DOC, *FILE, *IDP, *DSTID, *LMSG	Required, Positional 1
TOUSRID	Recipient	Values (up to 300 repetitions): *Element list*	Optional, Positional 2
	Element 1: User ID	*Character value*	
	Element 2: Address	*Character value*	
	Element 3: Recipient type	***PRI**, *CC, *BCC	
TOINTNET	Internet Recipient	Values (up to 300 repetitions): *Element list*	Optional
	Element 1: Internet address	*Character value*, ***NONE**	
	Element 2: Recipient type	***PRI**, *CC, *BCC	
DSTD	Description	*Character value*	Optional, Positional 3
MSG	Message	*Character value*, ***NONE**, *DSTIDMSG	Optional
LONGMSG	Long Message	*Character value*, ***NONE**	Optional
CFMDEL	Confirmation of delivery	***NO**, *YES	Optional
SENSITIV	Sensitivity	***NONE**, *PRIVATE, *PERSONAL, *CONFIDENTIAL	Optional
PERSONAL	Personal	***NO**, *YES	Optional
IMPORTANCE	Content importance	***NORMAL**, *LOW, *HIGH	Optional
PTY	Priority	***NORMAL**, *HIGH, *LOW	Optional
USRID	User identifier	Single values: ***CURRENT** Other values: *Element list*	Optional
	Element 1: User ID	*Character value*	
	Element 2: Address	*Character value*	
DOCFILE	Document file	Single values: ***NONE** Other values: *Qualified object name*	Optional
	Qualifier 1: Document file	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
DOCMBR	Document member	*Name*, ***FIRST**	Optional

Keyword	Description	Choices	Notes
DOCTYPE	Document type	2-65535, ***DFT**, *FFT, *RFT	Optional
SNDFMT	Send format	***NOCHG**, *NOTE, *FINALFORM	Optional
SYSCOD	System code	*Character value*, ***DFT**	Optional
DOCCHRID	Document character identifier	Single values: ***SYSVAL**, *DEVD Other values: *Element list*	Optional
	Element 1: Graphic character set	*Integer*	
	Element 2: Code page	*Integer*	
DSTID	Distribution identifier	*Character value*, ***NONE**	Optional
DSTIDEXN	Distribution ID extension	0-99, ***NONE**	Optional
DOC	Document	*Character value*, *DOCID	Optional
FLR	In folder	*Character value*, ***NONE**	Optional
DOCID	Document identifier	*Character value*, ***NONE**	Optional
IDPFILE	Profile file	Single values: ***NONE**, *DOCFILE, *DSTIDIDP Other values: *Qualified object name*	Optional
	Qualifier 1: Profile file	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
IDPMBR	Profile member	*Name*, ***FIRST**	Optional
DOCD	Document description	*Character value*, ***DFT**, *DSTD	Optional
AUTHOR	Author	Single values: ***NONE**, *USRID Other values (up to 50 repetitions): *Character value*	Optional
DOCCLS	Document class	*Character value*, ***NONE**	Optional
KWD	Keyword	Single values: ***NONE** Other values (up to 50 repetitions): *Character value*	Optional
SUBJECT	Subject	Single values: ***NONE**, *DOCD Other values (up to 50 repetitions): *Character value*	Optional
DOCDATE	Document date	*Date*, ***NONE**, *CURRENT	Optional
FILCAB	File cabinet location	*Character value*, ***NONE**	Optional
CPYLST	Copy list	Single values: ***NONE** Other values (up to 50 repetitions): *Character value*	Optional
EXPDATE	Expiration date	*Date*, ***NONE**	Optional
CRTDATE	Creation date	*Date*, ***NONE**, *CURRENT	Optional
REFERENCE	Reference	*Character value*, ***NONE**	Optional
ACTDATE	Action due date	*Date*, ***NONE**, *CURRENT	Optional
RPYDATE	Reply requested	*Element list*	Optional
	Element 1: Date	*Date*, ***NONE**, *CURRENT, *ANY	
	Element 2: Time	*Time*, ***ENDOFDAY**	
STATUS	Document status	*Character value*, ***NONE**	Optional
CMPDATE	Completion date	*Date*, ***NONE**, *CURRENT	Optional
PROJECT	Project	*Character value*, ***NONE**	Optional
DOCLANGID	Language ID	*Character value*, ***JOB**	Optional
DOCCNTRYID	Country or region ID	*Character value*, ***JOB**	Optional
ALWALTRCP	Allow alternate recipient	***YES**, *NO	Optional
DISCLOSE	Disclose recipient	***YES**, *NO	Optional
ALWX400CNV	Allow X.400 conversion	***YES**, *NO	Optional

Keyword	Description	Choices	Notes
AUTUSR	Authorizing user	Single values: ***NONE** Other values: *Element list*	Optional
	Element 1: User ID	*Character value*	
	Element 2: Address	*Character value*	
DSTEXPDATE	Distribution expiry indicator	*Element list*	Optional
	Element 1: Date	*Date*, ***NONE**	
	Element 2: Time	*Time*, ***ENDOFDAY**	
CMDCHRID	Command character identifier	Single values: ***SYSVAL**, *DEVD Other values: *Element list*	Optional
	Element 1: Graphic character set	*Integer*	
	Element 2: Code page	*Integer*	

Top

Information to be sent (TYPE)

Specifies the type of information that is sent and the parameters that are valid on this command.

***MSG** Only the message specified on the **Message** prompt (MSG parameter) is sent.

***DOC** The document specified on the **Document** prompt (DOC parameter) or the **Document identifier** prompt (DOCID parameter) is sent. The user must have authority for the document before it can be sent.

***FILE** The database file specified on the **Document file** prompt (DOCFILE parameter) and the **Document member** prompt (DOCMBR parameter) is sent. The database file is sent without any changes. The user must have authority for the database file before it can be sent.

***IDP** The interchange document profile (IDP) that is sent is specified on the **Profile file** prompt (IDPFILE parameter) and the **Profile member** prompt (IDPMBR parameter), or is specified in the document profile built by this command.

***DSTID**

The mail entry that is identified by the distribution ID is distributed. The distribution ID is called the distribution document name.

***LMSG**

The text specified on the Long Message prompt (LONGMSG parameter) is sent as an final-form text document (FFTDCA) note. LONGMSG allows up to 5000 characters, which is about one page of text. The ″details″ parameters (Subject, Reference, Author, Due Date, etc.) are supported with TYPE(*LMSG). The TOINTNET parameter is also supported with TYPE(*LMSG).

Top

Recipient (TOUSRID)

Specifies one of the following:

- The user ID and address of zero or more users to whom the distribution is being sent
- The distribution list name of zero or more distribution lists containing the user ID and address of one or more users to whom the distribution is being sent.

A combination of user IDs and distribution lists can be used on the same command. Up to 300 user IDs and addresses can be specified.

If no user ID and address are specified for the TOUSRID parameter, an internet address must be specified on the TOINTNET parameter.

The possible User ID or List ID values are:

user-ID
Specify the user ID of the user to whom the distribution is sent.

list-ID
Specify the distribution list of users to whom the distribution is sent.

The possible User Address or List Qualifier values are:

user-address
Specify the user address of the user (specified in Element 1) to whom the distribution is sent.

list-qualifier
Specify the distribution list qualifier of users (specified in Element 1) to whom the distribution is sent.

The possible Recipient Type values are:

***PRI** The user or distribution list is the primary recipient of the distribution.

***CC** The user or distribution list is receiving a copy of the distribution sent to the primary recipient. However, this copy recipient is not identified on the distribution as a receiver on the distribution.

***BCC** The user or distribution list is receiving a copy of the distribution. However, this copy recipient is not identified on the distribution as a receiver on the distribution.

Top

Internet Recipient (TOINTNET)

Specifies the internet address of one or more users to whom the distribution is being sent. If TOINTNET(*NONE) is specified, at least one user ID and address must be specified on the TOUSRID parameter. Up to 300 internet addresses can be specified.

The TOINTNET parameter is not allowed when the TYPE keyword is *MSG.

***NONE**
No internet address is specified. This is a single value.

character-value
Specify the internet address of a person or organization to whom the distribution is being sent. Up to 253 characters can be specified for each address.

The possible Recipient Type values are:

***PRI** The internet address is the primary recipient of the distribution.

***CC** The internet address will receive a copy of the distribution sent to the primary recipient.

***BCC** The internet address will receive a copy of the distribution. However, this copy recipient is not identified on the distribution as a receiver on the distribution.

Top

Description (DSTD)

Specifies the description of the distribution. A maximum of 44 characters can be specified. This parameter is required and can not start with a blank character or be all blank characters.

Top

Message (MSG)

Specifies whether a short message is sent with the distribution.

***NONE**
No message is sent.

***DSTIDMSG**
The message in the distribution document specified on the **Distribution identifier** prompt (DSTID parameter) is sent with the distribution.

message-text
Specify the message (256 characters maximum) to send to the users.

Top

Message (LONGMSG)

Specify the text to send to the recipients as an final-form text document (FFTDCA).

***NONE**
No text is sent. This is the default and required when TYPE is not *LMSG

long-message-text
Specify the text (5000 characters maximum) to send to the recipients. The text will be put into an final-form text document (FFTDCA) that will be sent as a note.

The text must be formatted by using the following controls within the text.

- :/N - Will cause a new line (Carrier return).
- :/P - Will cause a new paragraph. New line plus a blank line. (Carrier return and Required carrier return)

The document is setup with 10 characters to the inch, the font ID is 11, the left margin at position 1, and the right margin at position 75. If :/N is not used to start a new line, the text will continue past the right margin and will not be viewable by some E-mail clients. Blanks are not removed from the text, but left in place for indentation and alignment. No extra text (ie. Recipient List, Date/Time, Subject, or Sender) is put into the document.

Technical note to programmers: Any of the FFTDCA controls (in hexadecimal) can be inserted in the text to provide more control of the format if the controls above are not enough.

Top

Confirmation of delivery (CFMDEL)

Specifies whether the sender receives a confirmation of delivery notification when each receiver gets the distribution. Even though confirmation of delivery is not requested, the sender is still informed when the distribution is not delivered because of an user ID is not valid, a system failure, or a routing failure. If confirmation of delivery is requested, the sender is informed when the receiver receives, deletes, or runs another command against the distribution. To get this information, the sender must use the Query Distribution (QRYDST) command with *OUT specified on the **Incoming or outgoing** prompt (OPTION parameter).

***NO** Confirmation of delivery is not requested.

***YES** Confirmation of delivery is requested.

Top

Sensitivity (SENSITIV)

Specifies the level of sensitivity defined by the X.400 standard. The four levels include normal, personal, private and company confidential. Private distributions cannot be viewed by a user working on behalf of another user.

***NONE**
The distribution has no sensitivity restrictions.

***PERSONAL**
The distribution is sent to the recipient as an individual.

***PRIVATE**
The distribution contains information that should be accessed only by the recipient.

***CONFIDENTIAL**
The distribution contains information that should be handled according to company procedures.

Top

Personal (PERSONAL)

Specifies whether the document distribution is private or not. This parameter is replaced by SENSITIV but the PERSONAL parameter can still be used. However, because this parameter may be removed in a later release, whenever possible use the SENSITIV parameter.

If PERSONAL(*YES) is used, the SENSITIV parameter must be omitted or it must be with the value SENSITIV(*NONE). If the command is prompted without this parameter specified, this parameter is not displayed.

***NO** Only the owner and users that have authorization to the distribution document can get access to distributions that are not sensitive. Users authorized to work on behalf of other users who have access to the distribution can access documents that are not sensitive. This value will map to SENSITIV(*NONE).

***YES** Only the owner can get access to private distribution documents. Users authorized to work on behalf of other users who have access to the distribution document cannot get access to the distribution. This value will map to SENSITIV(*PRIVATE).

Top

Content importance (IMPORTANCE)

Specify low, normal or high importance. This is an indication to the recipient of the content importance of the distribution. It is different from priority which relates to the speed with which the distribution was sent.

***NORMAL**
A distribution of normal importance.

***HIGH**
A very important distribution.

***LOW** A low importance distribution.

Top

Priority (PTY)

Specifies whether the distribution is sent using low priority, normal priority or high priority. For distributions to remote receivers, the priority determines which ’SNADS Next System Queue’ is used. Normal priority distributions use the normal next system queue. Low priority distributions use the normal next system queue with the lowest priority available within that queue. High priority uses the priority next system queue. The difference between low, high and normal priority depends on information, specified by the user, on the Configuration Distribution Services (CFGDSTSRV) command. The handling of priority distributions by other office system nodes can vary, but generally the high priority distributions take the faster path when there is a choice of paths. For distribution to local receivers, the priority determines whether a message is sent to the receiver’s message queue to notify the receiver of the distribution. No message is sent for normal distributions.

***NORMAL**
Normal priority is used.

***HIGH**
High priority is used.

***LOW** Low priority is used.

Top

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.

Top

Document file (DOCFILE)

Specifies the names of the database file and the library that contains the document data. The database file is a user-defined file or the output file specified in either the Receive Distribution (RCVDST) command or the Retrieve Document (RTVDOC) command. If an output file is specified, only the data portion of the document data record is read from the output file. The prefix is removed from the document data record.

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 database file. If no library is specified as the library for the job, QGPL is used.

library-name
Specify the library where the database file is located.

data-base-file-name
Specify the name of the database file that contains the document data.

Top

Document member (DOCMBR)

Specifies the document database file member that is used.

***FIRST**
The first member created in the database file is used.

member-name
Specify the name of the database file member that is used.

Top

Document type (DOCTYPE)

Specifies the type of document being used. This identifier is used by the system to determine whether the data stream can be handled properly.

***DFT** The system creates the proper document type identifier based on the source of the data.

***FFT** The document is in Final Form Text. This type of document is intended to be viewed and printed, but not edited, by the receiver.

***RFT** The document is in Revisable Form Text. This type of document can be viewed, printed, and edited by the receiver.

document-type-number
Specify a document type identifier value ranging from 2 through 65,535. The numbers from 2 through 32,767 are controlled by registering them with the IBM Document Interchange Architecture and are used for IBM-defined document types. The numbers ranging from 32,768 through 65,535 are not registered with IBM and can be used for non-IBM-defined document types. The meaning of these document types must be determined by defining the value of the system code on the **System code** prompt (SYSCOD parameter).

Top

Send format (SNDFMT)

Allows the user to specify the format of the document being sent.

***NOCHG**
The document is sent in the current format.

***NOTE**
The document is sent in a final form text document content architecture (FFTDCA) data stream as a note.

***FINALFORM**
The document is sent in FFTDCA.

Top

System code (SYSCOD)

Specifies the text used with the value specified on the **Document type** prompt (DOCTYPE parameter) to help uniquely identify the type of document being used. The receiver of the data stream determines the document data stream and processing requirements to edit, view, print, or change the document.

***DFT** The system supplies a default system code. If the value specified on the **Document type** prompt (DOCTYPE parameter) is a number ranging from 2 through 32,767, the default is ’IBM System i5 CL’ and is retrieved from message CPX9026. If the value specified on the **Document type** prompt (DOCTYPE parameter) is in the range from 32,768 through 65,535, a system code must be specified.

system-code
Specify the text that uniquely identifies the type of document being sent. A maximum of 13 characters can be specified.

Top

Document character identifier (DOCCHRID)

Specifies the character identifier (graphic character set and code page) for the document data being used. The character identifier is related to the display device used to create the document data.

***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.

graphic-character-set code-page
Specify the graphic character set and code page values used to create the data being distributed.

Note: Both parts can be up to 5 digits in length.

Top

Distribution identifier (DSTID)

Specifies the unique distribution identifier of the distribution. The distribution identifier is assigned by the system when the distribution is originated. Distribution identifiers can be found by using the Query Distribution (QRYDST) command. Identifiers are also returned from the Send Distribution (SNDDST) command.

***NONE**
No distribution identifier is used.

distribution-id
Specify the 3-part distribution identifier which is composed of the second part of the sender’s user ID (padded on the right to 8 characters), the first part of the sender’s user ID (padded on the right to 8 characters), and a 4-digit zoned sequence number with leading zeros. For example, ’NEWYORK SMITH 0204’. This parameter is required when *DSTID is specified on the **Information to be sent** prompt (TYPE parameter).

Top

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 2-digit extension has a value ranging from 00 through 99 that uniquely identifies duplicate distributions. For incoming distributions, this extension ranges from 01 through 99. For confirmation of delivery distributions, this extension must be 00.

***NONE**
There is no duplicate distribution. *NONE is equivalent to an extension of 01.

distribution-id-extension
Specify the extension associated with the distribution. This is used to uniquely identify duplicate distributions.

Top

Document (DOC)

Specifies the name of the document being sent.

document-name
Specify the user-assigned name (10 characters maximum) of the document to be sent.

***DOCID**
The document being sent is identified by the library-assigned document name.

Top

In folder (FLR)

Specifies the name of the folder that contains the document being sent. This is the user-assigned name given to the folder when it is created. If document name is specified on the **Document** prompt (DOC parameter), then *DOCID must not be specified.

***NONE**
The document is not located in a folder.

folder-name
Specify the name of the folder that contains the document being sent. A folder name can consist of a series of folder names if the document being sent is located in a folder contained within another folder or folders.

Top

Document identifier (DOCID)

Specifies the library-assigned name of the document. This is the name assigned to the document by the system when it was created. Documents filed outside the local system have only library-assigned document names. The library-assigned document names can be determined by using the Query Document Library (QRYDOCLIB) command or by the message returned from the File Document (FILDOC) command.

Library-assigned document names are 24 characters in length with the following format:

```
YYYYMMDDHHMNSSHSSNSNSNSN
```

where:

```
      YYYY = year
        MM = month
        DD = day
        HH = hour
        MN = minute
        SS = second
        HS = hundredths of a second
   SNSNSNSN = system name
```

***NONE**
No library-assigned document name is required when the document is identified on the **Document** prompt (DOC parameter).

library-assigned-document-name
Specify the library-assigned name of the document being sent.

Top

Profile file (IDPFILE)

Specifies where the document profile information is located. If you specify this parameter, the remaining parameters after the **Profile member** prompt (IDPMBR parameter) are ignored, except the **Command character identifier** prompt (CMDCHRID parameter) and the **Document character identifier** prompt (DOCCHRID parameter).

***NONE**
The interchange document profile (IDP) is supplied by other parameters on this command. There is no database file containing the IDP information. If *NONE is specified, the **Profile member** prompt (IDPMBR parameter) is ignored.

***DSTIDIDP**
The IDP information associated with the distribution document is used. The **Profile member** prompt (IDPMBR parameter) is ignored. This is valid only when TYPE (*DSTID) is specified.

***DOCFILE**
The database file specified for the document also contains the profile information. If *DOCFILE is specified, the **Document file**prompt (DOCFILE parameter) and **Document member** prompt (DOCMBR parameter) are used for the document profile information.

data-base-file-name
Specify the name of the database file that contains the IDP. The document profile database file can be a user-defined file or the output file specified on the Receive Distribution (RCVDST) or Retrieve Document (RTVDOC) commands. If you specify a user-defined file, it must have the same format as the output file produced by RCVDST or RTVDOC. If an output file is specified, only the data portion of the document profile record is read from the output file. The prefix is removed from the document profile record.

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 database file. If no current library is specified as the library for the job, QGPL is used.

library-name
Specify the library where the database file is located.

Top

Profile member (IDPMBR)

Specifies the interchange document file member name being used. This parameter is used only when a database file name is also specified on the **Profile file** prompt (IDPFILE parameter).

***FIRST**
The first member created in the database file is used.

member-name
Specify the name of the database file member being used.

Top

Document description (DOCD)

Specifies a description for the document being distributed. This information is in the Document Interchange Architecture IDP document name field.

***DFT** The system generates a document description. For database files, the default format is library-name/file-name/member-name. For a description of a hard copy document, the default is the distribution description. For a description of a distribution document, the default is the document description associated with the distribution.

***DSTD**
The distribution description specified on the **Description** prompt (DSTD parameter) is used for the document description.

document-description
Specify the description of the document. A maximum of 44 characters can be specified.

Top

Author (AUTHOR)

Specifies the author or authors of the document.

You can enter multiple values for this parameter.

***NONE**
No author is identified for the document.

***USRID**
The user ID and address specified on the USRID parameter **User identifier** prompt(USRID parameter) is used as the author's name.

document-author-name
Specify the name of the author or authors. A maximum of 50 authors can be specified.

Top

Document class (DOCCLS)

Specifies the class associated with this document, such as MEMO, FORM, or SHEET.

***NONE**
No class is assigned to the document.

document-class
Specify the document class. A maximum of 16 characters can be specified.

Top

Keyword (KWD)

Specifies the keywords that describe the document.

You can enter multiple values for this parameter.

***NONE**
No keywords are defined for this document.

document-keyword
Specify the keywords to describe the document. A maximum of 50 keywords can be specified. Each keyword can have a maximum of 60 characters.

Top

Subject (SUBJECT)

Specifies the subject or subjects of the document.

You can enter multiple values for this parameter.

***NONE**
No subject is defined for the document.

***DOCD**
The document description is used as the subject for the document.

document-subject
Specify the subject or subjects of the document. A maximum of 50 subjects can be specified and each subject can have a maximum of 60 characters of text.

Top

Document date (DOCDATE)

Specifies any date the user needs to assign to the document.

***NONE**
No date is assigned to the document.

***CURRENT**
The system assigns the current system date to the document.

document-date
Specify the document date. The date must be specified in the job date format.

Top

File cabinet location (FILCAB)

Specifies the location of the document. This parameter is intended to describe the location of printed documents. The interchange document profile (IDP) that refers to the printed document is distributed. This parameter is required if *IDP is also specified on the **Information to be sent** prompt (TYPE parameter) and *NONE is specified on the **Profile file** prompt (IDPFILE parameter).

***NONE**
No filing cabinet reference is defined for this document.

filing-cabinet-reference
Specify the text that describes where the printed document is located. A maximum of 60 characters can be specified.

Top

Copy list (CPYLST)

Specifies the names of the users that receive this document.

You can enter multiple values for this parameter.

***NONE**
No copy list is included for this document.

recipient-list
Specify the names of the users that receive the document. A maximum of 50 names can be specified. Each name can have a maximum of 60 characters.

Top

Expiration date (EXPDATE)

Specifies the date on which the document is no longer needed.

***NONE**
No document expiration date is specified.

expiration-date
Specify the document expiration date. The date must be specified in the job date format.

Top

Creation date (CRTDATE)

Specifies the date the document was created.

***NONE**
No document creation date is specified.

***CURRENT**
The current system date is used as the date the document was created.

create-date
Specify the document creation date. The date must be specified in the job date format.

Top

Reference (REFERENCE)

Specifies a reference associated with the document.

***NONE**
No reference field is included for this document distribution.

reference
Specify text that describes the reference associated with the document. A maximum of 60 characters can be used.

Top

Action due date (ACTDATE)

Specifies the due date for the requested action. If no action due date is specified and a reply request date is specified the action due date is set from the reply request date.

***NONE**
No action due date is specified.

***CURRENT**
The current date is used.

action-due-date
Specify the value used as the action due date. The date must be specified in the format specified by the system value QDATFMT.

Top

Reply requested (RPYDATE)

Specifies the date and time a reply to the distribution is requested. This applies only to primary recipients.

The possible Reply by date values are:

***NONE**
No reply is required.

***CURRENT**
The current date is used. The time defaults to *ENDOFDAY and is set to 23:59:59.

***ANY** A reply is requested but no date and time are specified.

reply-by-date
Specify the value used as the reply requested date. The date must be specified in the format specified by the system value QDATFMT.

The possible Reply by time values are:

***ENDOFDAY**
A reply is requested by the end of the specified date. The time is set to 23:59:59.

reply-by-time
Specify the value used as the reply requested time.

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 digits (hours and minutes, or hours, minutes, and seconds). The time separator specified by the system value QTIMSEM is used to separate the hours, minutes, and seconds. If the command is entered from the command line, the string must be entered in apostrophes. If a time separator other than the separator used for your job is used, the command fails. Valid values for hh range from 00 to 23. Valid values for mm and ss range from 00 to 59.

Top

Document status (STATUS)

Specifies the user-defined status of the document. Examples of status are: In Process, Pending Approval, or Retired.

***NONE**
No status is included in this document.

status-of-document
Specify text that describes the status of the document. A maximum of 20 characters can be specified.

Top

Completion date (CMPDATE)

Specifies the date when the requested action is completed.

***NONE**
No completion date is included.

***CURRENT**
The current system date is used as the completion date.

date-complete
Specify the action completion date. The date must be specified in the job date format.

Top

Project (PROJECT)

Specifies the project associated with the document.

***NONE**
No project field information is included in this document.

project
Specify text that describes the project of the document. A maximum of 10 characters can be specified.

Top

Language ID (DOCLANGID)

Specifies the language identifier to be placed in this document's interchange document profile (IDP).

Note: This parameter is ignored if the **Profile file** prompt (IDPFILE parameter) is specified, or if *MSG or *DOC is specified on the **Information to be sent** prompt (TYPE parameter).

***JOB** The language identifier specified for the job in which this command is entered is used.

language-identifier
Specify a language identifier. Press the F4 key from the **Language ID** prompt (DOCLANGID parameter) to see a list of valid identifiers.

Top

Country or region ID (DOCCNTRYID)

Specifies the country or region identifier to be placed in this document’s interchange document profile (IDP).

Note: This parameter is ignored if the **Profile file** prompt (IDPFILE parameter) is specified, or if *MSG or *DOC is specified on the **Information to be sent** prompt (TYPE parameter).

***JOB** The country or region identifier specified for the job in which this command is entered is used.

country-or-region-identifier
Specify a country or region identifier. Press the F4 key from the **Country or region ID** prompt (DOCCNTRYID parameter) to see a list of valid identifiers.

Top

Allow alternate recipient (ALWALTRCP)

Specify if the distribution can be delivered to an alternate recipient, determined by the receiving system. This function is used only by X.400 network systems. This function provides the ability to direct certain mail to a specific user.

The default of this field is *YES. Because the user might change the sensitivity field to private but might not change the alternate recipient field to no, The user might send a private item to an alternate recipient. No error message is generated for this condition.

***YES** Specifies the distribution can be delivered to an alternate recipient.

***NO** Specifies the distribution cannot be delivered to an alternate recipient.

Top

Disclose recipient (DISCLOSE)

Specifies whether or not each recipient gets a list of the other recipients.

***YES** Disclose recipients.

***NO** Do not disclose recipients.

Top

Allow X.400 conversion (ALWX400CNV)

Specifies whether an X.400 conversion is allowed on the distribution being submitted. In certain circumstances, the end user knows the capabilities of the recipient and may indicate that the distribution is not to be converted. The system administrator can define whether or not conversion is to take place. This field allows the user to override that setting on a message by message basis. For example, if the gateway allows conversion, the user could specify that conversion is to be prohibited. This will take precedence over the gateway setting. The distribution would then not be converted.

Note: This field is valid only for X.400 support. This field will not affect System i5 data stream transformations, such as RFT to FFTDCA.

***YES** Conversion may be performed on the distribution by the receiving system.

***NO** No conversion is permitted on the distribution by the receiving system.

Top

Authorizing user (AUTUSR)

Specifies the user ID and address of the user that authorized the content of this distribution. The authorizing user will receive a copy of the distribution.

***NONE**
No authorizing user.

The possible User ID value is:

user-ID
Specify the user ID of the user from whom the distribution is authorized.

The possible User Address value is:

user-address
Specify the user address of the user from whom the distribution is authorized.

Top

Distribution expiry indicator (DSTEXPDATE)

Specifies the date and time on which the distribution is no longer needed in the mail log.

The possible Distribution Expiration Date values are:

***NONE**
The distribution has no expiration date.

dist-expiration-date
Specify the value to use as the expiration date for the distribution. The date must be in the format specified by the system value QDATFMT.

The possible Distribution Expiration Time values are:

***ENDOFDAY**
An expiration time is requested by the end of the specified date. The time is set to 23:59:59.

dist-expiration-time
Specify the value as the expiration time.

The time is specified in a 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 digits (hours and minutes, or hours, minutes, and seconds). The time separator specified by the system value QTIMSEM is used to separate the hours, minutes, and seconds. If the command is entered from the command line, the string must be enclosed in apostrophes. If a time separator other than the separator used for your job is used, the command fails. Valid values for hh range from 00 to 23. Valid values for mm and ss range from 00 to 59.

Top

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.

In some cases, the data is translated to a code page and character set that is interchangeable with other IBM OfficeVision/400 products. The interchangeable character set and code page is ’697 500’, except for the **User identifier** prompt (USRID parameter), **Recipient** prompt (TOUSRID parameter), and **Distribution identifier** prompt (DSTID parameter), for which it is ’930 500’. In other cases, the code page and character set are attached to the field and sent with the field to allow the receiving terminal to correctly print and display the field.

The following parameters are translated:

- **Recipient** (TOUSRID)
- **Internet Recipient** (TOINTNET)
- **User identifier** (USRID)
- **Distribution identifier** (DSTID)
- **Document system code** (SYSCOD)
- **Message** (MSG)
- **Description** (DSTD)

The code page and character set is attached to the following parameters:

- **Long Message** (LONGMSG)
- **Document description** (DOCD)
- **Author** (AUTHOR)
- **Document class** (DOCCLS)
- **Keyword** (KWD)
- **Subject** (SUBJECT)
- **File cabinet location** (FILCAB)
- **Copy list** (CPYLST)
- **Reference** (REFERENCE)
- **Document status** (STATUS)
- **Project** (PROJECT)

Single values

***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.

Top

Examples

Example 1: Sending a Distribution on Behalf of Another User

```
SNDDST   TYPE(*FILE)  TOUSRID((JACKSON RCH38DB))
         DOCTYPE(20000)  SYSCOD(BRANDX)
         DOCFILE(DEPT46ELIB/XTEXT)  DOCMBR(GOLD1IPFS)
         PTY(*HIGH)  USRID(JACOBSON RCH38NBS)
         DSTD('IPFS FOR GOLD1 PROJECT')  CFMDEL(*YES)
         MSG('Update section 1.2.4. Return for final printing')
```

This command sends a distribution that is being sent by someone (such as a secretary) who is authorized to work on behalf of JACOBSON. The document being sent is a BRANDX text document that is sent to another user who also has the BRANDX text processor.

Example 2: Sending a Mail Log Entry

```
SNDDST   TYPE(*DSTID)  DSTID('NEWYORK SMITH 0204')
         DSTIDEXN(02)  TOUSRID((JACKSON RCH38DB))
         MSG(*DSTIDMSG)  CFMDEL(*YES)
```

This command sends a mail log entry that is identified by the distribution document name **NEWYORK SMITH 0204** that is distributed to user JACKSON at address RCH38DB. The message in the distribution document is distributed with the distribution.

Top

Error messages

***ESCAPE Messages**

CPF8A87
Document name &2 not correct.

CPF8A97
Folder name &1 not correct.

CPF89AA
*FINALFORM for send format (SNDFMT) valid only for *DOC for type (TYPE).

CPF89AB
Specify *IDP or *MSG for type (TYPE) only with *NOCHG for send format (SNDFMT).

CPF900B
User ID and address &1 &2 not in System Distribution Directory.

CPF900C
Sign on and verify of user failed.

CPF901A
Send distribution request failed.

CPF903D
Incorrect document identifier specified.

CPF905C
Error occurred trying to find a translation table.

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.

Top

Send Distribution Queue (SNDDSTQ)

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

Parameters
Examples
Error messages

The Send Distribution Queue (SNDDSTQ) command is used:

- To send a distribution queue's entries when the distribution queue is configured to be manually started but no operator is available.
- To override any distribution queue scheduling attributes and begin sending a queue's entries immediately.
- To restart a SNADS sender job that failed abnormally.

The SNDDSTQ command is primarily intended for use in a batch CL program. The SNDDSTQ command enables the same functions as option 2 (Send distribution queue) on the Work with Distribution Queue (WRKDSTQ) command main list panel. The SNDDSTQ command allows the functions to be started from a batch job instead of interactively.

Distribution queue names are translated to the graphic character set and code page 930 500, using the job's coded character set identifier (CCSID).

Restrictions:

- This command is shipped with public *EXCLUDE authority, and the QPGMR and QSYSOPR user profiles have private authorities to use the command.
- Messages that report errors about distribution queues may display or print different characters than you entered for the distribution queue name because of internal system transformations. Similarly (depending on the language used for the work station), the internal value for a distribution queue name may differ from the characters shown for the Work with Distribution Queue (WRKDSTQ) command. An error may be reported if the character-string value specified for the **Distribution queue** prompt (DSTQ parameter) does not match the rules for an internal distribution queue value or if it does not match the internal value for any defined distribution queue (ignoring case differences).

Top

Parameters

Keyword	Description	Choices	Notes
DSTQ	Distribution queue	*Character value*	Required, Positional 1
PTY	Priority	*NORMAL, *HIGH	Required, Positional 2

Top

Distribution queue (DSTQ)

Specifies the name of the distribution queue that is sent. The queue must have been previously configured using the Configure Distribution Services (CFGDSTSRV) command or the Add Distribution Queue (ADDDSTQ) command.

This is a required parameter.

Top

Priority (PTY)

Specifies whether the normal priority or high priority portion of the specified queue is sent.

The possible values are:

***NORMAL**
The normal priority queue is for those distributions with a service level of data low.

***HIGH**
The high priority queue is for those distributions with a service level of fast, status, or data high.

Note: This value is not valid for a SystemView distribution services (SVDS) type of distribution queue.

This is a required parameter.

Top

Examples

Example 1: Sending Distributions with Normal Priority

```
SNDDSTQ   DSTQ(CHICAGO)  PTY(*NORMAL)
```

This command sends distributions from the normal priority portion of the CHICAGO distribution queue.

Example 2: Sending Distributions with High Priority

```
SNDDSTQ   DSTQ(ATLANTA)  PTY(*HIGH)
```

This command sends distributions from the high priority portion of the ATLANTA distribution queue.

Top

Error messages

***ESCAPE Messages**

CPF8802
Distribution queue &1 was not found.

CPF8805
Special value for System name/Group not permitted or not used correctly.

CPF8806
Value &1 not valid for system name or system group.

CPF881C
High priority queue not allowed for *SVDS distribution queue &1

CPF8812
Error occurred while processing distribution queues.

CPF8816
QSNADS communications subsystem is not active.

CPF8817
Distribution queue is held.

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.

Top

Send File (SNDF)

Where allowed to run:

- Batch program (*BPGM)
- Interactive program (*IPGM)

Threadsafe: No

Parameters
Examples
Error messages

The Send File (SNDF) command is used by a CL program or ILE CL procedure to send a record to a display device that is being used by an interactive user. The device can be any display station, including the console. The command sends the data from the program's CL variables to the display's device file in the specified record format. These variables were automatically declared in the program (one for each field in the record format) when the CL source program was compiled and a Declare File (DCLF) command was processed as part of the source.

Of the record formats specified in the DCLF command, only one can be specified in each SNDF command. If the device file has not been opened, it is opened by this command. The file and record format specified in this command can be overridden by an Override with Display File (OVRDSPF) command if it is entered before the file is opened. However, care should be taken that the fields in the overriding record format correspond to the CL variables declared in the program.

Restrictions:

- This command is valid only within a CL program or ILE CL procedure.
- This command is valid only for display files.
- This command cannot be used with database files.

Top

Parameters

Keyword	Description	Choices	Notes
DEV	Display device	*Name*, ***FILE**	Optional, Positional 1
RCDFMT	Record format	*Name*, ***FILE**	Optional, Positional 2
OPNID	Open file identifier	*Simple name*, ***NONE**	Optional

Top

Display device (DEV)

Specifies the name of the display device to which the data in the CL variables for the specified record format is to be sent.

***FILE** The program's data is to be sent to the device associated with the device file that was declared in the FILE parameter of the DCLF command. If more than one device name is specified in the device file, *FILE cannot be specified.

name Specify the name of the device or the name of the CL variable that contains the name of the device to which the program's data is to be sent.

Top

Record format (RCDFMT)

Specifies the name of the record format that is to be used to send data to the file. The format contains all the fields in the record. This parameter must be coded with a record format name if there is more than one record format name in the device file; *FILE cannot be coded if there is more than one. If the record format contains the INVITE DDS keyword (optioned on), the SNDF functions as if SNDRCVF WAIT(*NO) had been coded.

<u>*FILE</u> There is only one record format in the device file; that is the format in which the program's data is to be sent to the file.

name Specify the name of the record format in which the program's data is to be sent to the file. A CL variable cannot be used to specify the record format name.

Top

Open file identifier (OPNID)

Specifies the open file identifier that was declared on a preceding Declare File (DCLF) command in the same CL program or ILE CL procedure. A CL variable cannot be specified for this parameter value.

<u>*NONE</u>
No open file identifier is provided. This command will use the file associated with the DCLF command that had *NONE specified for the OPNID parameter. Only one file can be declared in a CL program or ILE CL procedure with *NONE as the open file identifier.

simple-name
Specify a name that matches the OPNID parameter value on a preceding DCLF command in the same CL program or ILE CL procedure.

Top

Examples

Example 1: Using Display File with One Record Format

```
DCLF   FILE(MENU1)
  :
SNDF
```

The record format in the device file MENU1 is sent to the device specified in the file. There is only one record format in the file.

Example 2: Using Display File with Multiple Record Formats

```
DCLF   FILE(SCREEN1)  RCDFMT(REC1 REC2)
  :
SNDF   DEV(DISP3)  RCDFMT(REC1)
```

The device file named SCREEN1 causes the display station named DISP3 to display the data sent by the CL program or ILE CL procedure. The data is shown in the format specified by the REC1 record format.

Example 3: Using Open File Identifier

```
DCLF   FILE(SCREEN1)  RCDFMT(REC1 REC2)  OPNID(OUTDSP1)
DCLF   FILE(SCREEN2)  RCDFMT(REC3 REC4)  OPNID(OUTDSP2)
 :
SNDF   DEV(*FILE)  RCDFMT(REC2)  OPNID(OUTDSP1)
```

The device file named SCREEN1 is used to send data to the display device named in the same device file. The data is presented to the user in the format specified by record format REC2. The SNDF command is associated with device file SCREEN1 because the open file identifier specified on the SNDF command matches the open file identifier specified on the DCLF command for display file SCREEN1.

Top

Error messages

***ESCAPE Messages**

CPF0859
File override caused I/O buffer size to be exceeded.

CPF0861
File &1 in library &2 is not a display file.

CPF0864
End of file detected for file &1 in &2.

CPF0883
*FILE not valid in DEV parameter for file &1.

CPF0887
Data available from previous input request.

CPF4101
File &2 in library &3 not found or inline data file missing.

CPF5068
Program device &4 not found in file &2 in library &3.

CPF5070
File &2 in library &3 has no program devices acquired.

Top

Send Journal Entry (SNDJRNE)

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

Parameters
Examples
Error messages

The Send Journal Entry (SNDJRNE) command is used to write a single journal entry to a specific journal. The entry can contain any information. The user may assign an entry type to the journal entry and may also associate the journal entry with a specified journaled object.

If the journal currently has a state of *STANDBY, then the journal entry will not be deposited unless OVRSTATE(*STANDBY) is specified.

The journal code for the entry is U, which indicates a user-specified journal entry.

Note: The Send Journal Entry (QJOSJRNE) Application Programming Interface (API) can also be used to write a user-specified journal entry to a specific journal. Using this API may improve performance and can provide additional function that is not available with this command. For more information, see the APIs topic collection in the Programming category in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/.

Restrictions:

- If an object other than a file is specified, it must be currently journaled to the specified journal.

  If a file object is specified, it must either be currently journaled to the specified journal or it must have been last journaled to the specified journal.
- The specified journal cannot be a remote journal.
- The specified journal cannot have a journal state of *INACTIVE.

Top

Parameters

Keyword	Description	Choices	Notes
JRN	Journal	*Qualified object name*	Required, Positional 1
	Qualifier 1: Journal	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
TYPE	Journal entry type	*Character value*, **00**	Optional, Positional 2
ENTDTA	Entry data	*Character value*, ***BLANK**	Optional, Positional 3
FILE	Journaled physical file	Single values: ***NONE** Other values: *Element list*	Optional
	Element 1: Physical file	*Qualified object name*	
	Qualifier 1: Physical file	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
	Element 2: Member	*Name*, ***FIRST**, *NONE	

Keyword	Description	Choices	Notes
OBJ	Object	Single values: ***NONE** Other values: *Element list*	Optional
	Element 1: Object	*Qualified object name*	
	Qualifier 1: Object	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
	Element 2: Object type	***FILE**, *DTAARA, *DTAQ, *LIB	
	Element 3: Member	*Name*, ***FIRST**, *NONE	
OBJPATH	Object	*Path name*, ***NONE**	Optional
OBJFID	File identifier	*Hexadecimal value*, ***NONE**	Optional
FORCE	Force journal entry	***NO**, *YES	Optional
OVRSTATE	Override journal state	***NONE**, *STANDBY	Optional

Top

Journal (JRN)

Specifies the journal to contain the new journal entry.

This is a required parameter.

Qualifier 1: Journal

journal-name
Specify the name of the journal to contain the new journal entry.

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.

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

Top

Journal entry type (TYPE)

Specifies the journal entry type of this journal entry.

00 The journal entry type is a ’00’ (hex F0F0).

entry-type
Specify a 2-character value or hex value used for the journal entry type. This value must be greater than or equal to hex C000.

If a hexadecimal value is specified that does not represent characters, that value is not shown on the DSPJRN display or on the printout.

Top

Entry data (ENTDTA)

Specifies the user-specified data that is placed in the variable portion of the journal entry.

***BLANK**
No user-specified data is placed in the journal entry.

'entry specific-data'
Specify up to 3000 characters, enclosed in apostrophes.

Top

Journaled physical file (FILE)

Specifies the database physical file and member with which this entry is associated.

If this parameter is specified, the parameters OBJ, OBJPATH or OBJFID cannot also be specified.

Single values

***NONE**
There is no associated physical file for this entry.

Element 1: Physical file

Qualifier 1: Physical file

file-name
Specify the name of the physical file with which this entry is associated.

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.

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

Element 2: Member

***FIRST**
The entry is associated with the first member in the file.

***NONE**
The entry is associated with the file, not with any member of the file.

member-name
Specify the name of the physical file member with which this entry is associated.

Top

Object (OBJ)

Specifies the object with which this entry is associated.

If this parameter is specified, the parameters FILE, OBJPATH or OBJFID cannot also be specified.

Single values

***NONE**
There is no associated object for this entry.

Element 1: Object

Qualifier 1: Object

object-name
Specify the name of the object with which this entry is associated.

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.

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

Element 2: Object type

***FILE** The entry is associated with a database file or database file member.

***DTAARA**
The entry is associated with a data area.

***DTAQ**
The entry is associated with a data queue.

***LIB** The entry is associated with a library.

Element 3: Member

***FIRST**
The entry is associated with the first member in the file.

***NONE**
The entry is associated with the file, not with any member of the file.

member-name
Specify the name of the physical file member with which this entry is associated.

Note: If the specified object type is not *FILE, the member name value is ignored.

Top

Object (OBJPATH)

Specifies the path-qualified object name with which this entry is associated. Only objects whose path name identifies an object of type *STMF, *DIR or *SYMLNK that is in the ″root″ (/), QOpenSys, and user-defined file systems are supported.

Note: This parameter is Unicode-enabled. See ″Unicode support in CL″ in the CL topic collection in the Programming category in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/ for additional information.

If this parameter is specified, the parameters FILE, OBJ or OBJFID cannot also be specified.

***NONE**
There is no associated object for this entry.

path-name
Specify the name of the object with which this entry is associated.

Top

File identifier (OBJFID)

Specifies the file-identifier (FID) with which this entry is associated. FIDs are a unique identifier associated with integrated file system related objects. This field is input in Hexadecimal format. Only objects whose FID identifies an object of type *STMF, *DIR or *SYMLNK that is in the ″root″ (/), QOpenSys, and user-defined file systems are supported.

If this parameter is specified, the parameters FILE, OBJ or OBJPATH cannot also be specified.

***NONE**
There is no associated object for this entry.

file-identifier
Specify the FID for the object with which this entry is associated.

Top

Force journal entry (FORCE)

Specifies whether the journal receiver is forced to auxiliary storage after the user entry is written to it.

***NO** The journal receiver is not forced to auxiliary storage.

***YES** The journal receiver is forced to auxiliary storage.

Top

Override journal state (OVRSTATE)

Specifies whether the journal entry will be deposited, overriding the current state of the journal.

***NONE**
None of the journal state values are overridden. That is,

- The journal entry is deposited if the journal state is *ACTIVE.
- The journal entry is not deposited and an error is not sent if the journal state is *STANDBY.

***STANDBY**
The journal entry is deposited even if the journal state is *STANDBY.

Top

Examples

Example 1: Forcing Journal Receivers to Auxiliary Storage

```
SNDJRNE   JRN(JRNLA)  TYPE(AB)  ENTDTA('PROGRAM COMPLETE')
          FILE(MYLIB/ORDERENT MBR1)  FORCE(*YES)
```

If the journal currently has a journal state of *ACTIVE, this command places a journal entry of type AB (hex C1C2) with the journal entry data ′PROGRAM COMPLETE′ in the current journal receivers attached to journal JRNLA as found by using the library search list. The entry is associated with member MBR1 of file ORDERENT in library MYLIB. The journal receiver is forced to auxiliary storage after the entry has been placed in it.

Example 2: Sending a Journal Entry

```
SNDJRNE   JRN(JRNLA)  TYPE(x'C1F1')  OVRSTATE(*STANDBY)
```

If the journal currently has a journal state of *STANDBY or *ACTIVE, this command places a journal entry of type ′A1′ (hex C1F1) with no journal entry data in the current journal receiver attached to journal JRNLA as found by using the library search list. The entry is not associated with any physical file member.

Top

Error messages

***ESCAPE Messages**

CPFA0D4
File system error occurred. Error number &1.

CPF7002
File &1 in library &2 not a physical file.

CPF7003
Entry not journaled to journal &1. Reason code &3.

CPF7007
Cannot allocate member &3 file &1 in &2.

CPF7037
Object not journaled to journal &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.

CPF9809
Library &1 cannot be accessed.

CPF9810
Library &1 not found.

CPF9812
File &1 in library &2 not found.

CPF9815
Member &5 file &2 in library &3 not found.

CPF9820
Not authorized to use library &1.

CPF9822
Not authorized to file &1 in library &2.

Top

Send Message (SNDMSG)

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

Parameters
Examples
Error messages

The Send Message (SNDMSG) command is used by a display station user to send an immediate message from his display station to one or more message queues. (An immediate message is a message that is not predefined and is not stored in a message file.) The message can be sent to the system operator, to other display station users, to a user's message queue, all currently active users' message queues or to the system history log, QHST. The sender can require a reply from the message receiver. The primary users of this command are display station users and the system operator.

Restrictions:

1. You must have object operational (*OBJOPR) and add (*ADD) authorities for the message queue.
2. You must have use (*USE) authority for the specified message queues and *USE authority for the libraries in which they are located.
3. The SNDMSG command only allows a message of up to 512 characters of first-level message text to be sent.
4. This command can only send inquiry messages (specified by MSGTYPE(*INQ)) to one message queue or to two message queues if one of the queues is *HSTLOG.

Top

Parameters

Keyword	Description	Choices	Notes
MSG	Message text	*Character value*	Required, Positional 1
TOUSR	To user profile	*Name*, *SYSOPR, *ALLACT, *REQUESTER	Optional, Positional 3
TOMSGQ	To message queue	Single values: *SYSOPR Other values (up to 50 repetitions): *Qualified object name*	Optional, Positional 2
	Qualifier 1: To message queue	*Name*, *HSTLOG	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
MSGTYPE	Message type	***INFO**, *INQ	Optional, Positional 4
RPYMSGQ	Message queue to get reply	Single values: ***WRKSTN** Other values: *Qualified object name*	Optional
	Qualifier 1: Message queue to get reply	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
CCSID	Coded character set ID	1-65535, *HEX, ***JOB**	Optional

Top

Message text (MSG)

Specifies the immediate message that is being sent. The text must be enclosed in apostrophes if it contains blanks or other special characters. A maximum of 512 characters can be specified.

Coded Character Set Identifier (CCSID) Considerations

The text supplied for the MSG parameter is assumed to be in the CCSID of the job running this command unless the coded character set identifier is supplied for the CCSID parameter. For more information about the message handler and its use of CCSIDs, see the i5/OS globalization topic collection in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/.

This is a required parameter.

Top

To user profile (TOUSR)

Specifies that the message is to be sent to the message queue specified in the user profile for the user named on this parameter. This parameter cannot be used if a value is specified for the **To message queue(TOMSGQ)** parameter.

Either this parameter or the **To message queue (TOMSGQ)** parameter is required.

user-profile-name
Specify the user profile name of the user to whom the message is sent.

***SYSOPR**
The message is sent to the system operator message queue, QSYS/QSYSOPR. Any message sent to QSYSOPR automatically has a copy of the message sent to QHST.

***REQUESTER**
The message is sent to the user profile message queue for interactive jobs or to the system operator's message queue (QSYS/QSYSOPR) for batch jobs.

***ALLACT**
A copy of the message is sent to the user profile message queue of each user profile with an interactive job currently running. *ALLACT cannot be specified with inquiry messages.

Top

To message queue (TOMSGQ)

Specifies one to fifty message queues to which an informational message is sent. For an inquiry message, one message queue may be specified or two message queues may be specified if one of the queues is *HSTLOG. This parameter cannot be used if a value is specified for the **To user profile (TOUSR)** parameter.

Either this parameter or the **To user profile (TOUSR)** parameter is required.

Single values

***SYSOPR**
The message is sent to the system operator message queue, QSYS/QSYSOPR. Any message sent to QSYSOPR automatically has a copy of the message sent to QHST.

Qualifier 1: To message queue

***HSTLOG**
The message is sent to the system history log message queue, QSYS/QHST. If *HSTLOG is specified more than once, only one message will be sent to QSYS/QHST. If *HSTLOG is specified with QSYSOPR only one message is sent to QSYS/QHST.

message-queue-name
Specify the name of the message queue to which the message is to be 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 job is used to locate the message queue. If no current library entry exists in the library list, QGPL is used.

library-name
Specify the library where the message queue is located.

Top

Message type (MSGTYPE)

Specifies the type of message to be sent. Only an informational or inquiry message can be specified.

***INFO**
An informational message is to sent.

***INQ** An inquiry message is to be sent. The message queue receiving the message can reply to it. Inquiry messages are sent to only one message queue at a time or a second queue can be specified for TOMSGQ if the value is *HSTLOG.

Top

Message queue to get reply (RPYMSGQ)

Specifies, only if an inquiry message is sent, the message queue to which a reply is sent.

Single values

***WRKSTN**
The reply to the message is sent to the display station message queue associated with the sender's display station.

Note: This value cannot be specified for batch jobs.

Qualifier 1: Message queue to get reply

message-queue-name
Specify the name of the message queue to which a reply is sent. Only a user message queue, a display station message queue, or the system operator message queue can be specified.

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 message queue. If no current library entry exists in the library list, QGPL is used.

library-name
Specify the library where the message queue is located.

Top

Coded character set ID (CCSID)

Specifies the coded character set identifier (CCSID) that the supplied message text is in. For more information about the message handler and its use of CCSIDs, see the i5/OS globalization topic collection in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/.

***JOB** The message text is assumed to be in the CCSID of the job running this command.

***HEX** The message text is not converted. CCSID 65535 is used.

coded-character-set-identifier
Specify a valid CCSID in which you want your message text to be considered in. Valid values are between 1 and 65535. This command validates the CCSID.

Top

Examples

Example 1: Sending Message to User Message Queue

```
SNDMSG   MSG('Do you want to update INV now?')  TOUSR(JONES)
         MSGTYPE(*INQ)  RPYMSGQ(SMITH)
```

This command sends a message to the user message queue JONES. When the message is answered, the reply will be sent to the message queue SMITH.

Example 2: Sending Message to System's History Log

```
SNDMSG   MSG('Errors on PAYROLL cost me 1 hour of run time.')
         TOMSGQ(QHST)
```

This command is used by the system operator to send an informational message to the system's history log, QHST, through the log's message queue, which has the same name.

Example 3: Sending Message to System Operator

```
SNDMSG   MSG('Please make 2 copies of file LABORSTAT.')
         TOMSGQ(QSYSOPR)
```

This command shows a typical use of the SNDMSG command by a display station user. The user is sending the message to the system operator.

Example 4: Sending Message that Requires a Reply

```
SNDMSG   MSG ('How long will the system be up today?')
         TOMSGQ(*SYSOPR)  MSGTYPE(*INQ)
```

This command sends an inquiry message to the system operator. The message requires a reply. The system operator displays the message by using the DSPMSG command and enters the reply on the display. The reply is then sent to the display station user's work station message queue. The display station user enters another DSPMSG command to display the reply.

Top

Error messages

***ESCAPE Messages**

CPF2428
Message queue parameter is not valid.

CPF2433
Function not allowed for system log message queue &1.

CPF2469
Error occurred when sending message&1.

CPF247E
CCSID &1 is not valid.

CPF2488
Reply message queue *WRKSTN not valid for batch job.

CPF9830
Cannot assign library &1.

CPF9838
User profile storage limit exceeded.

Top

Send Network File (SNDNETF)

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

Parameters
Examples
Error messages

The Send Network File (SNDNETF) command sends a save file or a member of a physical database file to another user on the local system or on a remote system through the SNADS network. This command can be used to:

- Send data files to a user.
- Send source files to a user. Source sequence information is kept in the file sent.
- Send other object types stored in a save file to a user.

When the file arrives at its destination, a notification message is sent to both the recipient and sender of the file.

When a source physical file is sent, the source sequence number and change date in positions 1 through 12 of the record are sent with the file. These are kept if the file is received into a source physical file, and are truncated if the file is received into a nonsource physical file. When a file that was originally a nonsource physical file is received into a source physical file, the source sequence numbers are created and placed in front of the records.

Note: Save files created on the System i5 cannot be distributed to System/38. However, save files created on System/38 **can** be distributed to the System i5.

This command does not perform any CCSID translation on the contents of the file. However, the user ID and address of both the recipient and the originator are translated from the current job CCSID to the multinational character set 697/500.

Restrictions:

1. The user must be enrolled in the system distribution directory.
2. The maximum size of a file that can be sent using the SNDNETF command is approximately 2 billion bytes.

Top

Parameters

Keyword	Description	Choices	Notes
FILE	File	*Qualified object name*	Required, Positional 1
	Qualifier 1: File	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
TOUSRID	User ID	Values (up to 50 repetitions): *Element list*	Required, Positional 2
	Element 1: User ID	*Character value*	
	Element 2: Address	*Character value*	
MBR	Member	*Name*, ***FIRST**	Optional, Positional 3
TOTYPE	To file type	***FROMFILE**, *DATA	Optional

Keyword	Description	Choices	Notes
CLASS	VM/MVS class	**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	Optional
FORMAT	Send format	***V**, *F, *U	Optional
PTY	Send priority	***NORMAL**, *HIGH	Optional

Top

File (FILE)

Specifies the name and library of the file that is sent. The file being sent can be a physical file or a save file; logical files and device files are not allowed. Overrides to the specified file are ignored.

This is a required parameter.

The possible library values are:

***LIBL** The library list 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, QGPL is used.

library-name
Specify the library where the file is located.

Top

User ID (TOUSRID)

Specifies the two-part user ID of one or more users to whom the file is being sent, or the name of one or more distribution lists containing the user IDs of one or more users to whom the file is being sent. A combination of user IDs and distribution lists can be specified on the same command. Each user ID or distribution list is specified as a two-part name, and both parts are required.

This is a required parameter.

You can enter multiple values for this parameter.

Top

Member (MBR)

Specifies the member that is sent from the file. A member name is not allowed if the file is a save file.

***FIRST**
The first member (in order of creation) in the file is sent.

member-name
Specify the name of the file member that is sent.

Top

To file type (TOTYPE)

Specifies, when you send a source file, whether the sequence numbers and date fields are to be removed from the transmitted copy of the file. The source file is not changed. This parameter is not valid for non-source files.

The possible values are:

***FROMFILE**
The file type of the source file is used when sending the file. The transmitted file does not change.

***DATA**
The file is sent as a non-source file. The transmitted copy is sent without sequence numbers and date fields.

Top

VM/MVS class (CLASS)

Specifies the VM/MVS SYSOUT class for distributions sent to a VM host system or to an MVS host system.

The possible values are:

A The class is A.

B-Z, 0-9
Specify the distribution class. Valid values range from B through Z and 0 through 9.

Top

Send format (FORMAT)

Specifies the record format in which the network file is transmitted.

The possible values are:

***V** The file is sent using variable-length records with trailing blanks removed from each record.

***F** The file is sent as fixed-length records with no trailing blanks removed from the records. Specifying this value affects only network files sent to a System/370. This value is not recommended for sending files to another System i5.

***U** If the file contains null capable fields, specifying this value causes SNDNETF to allow sending of the file and to send the mapping information which indicates which fields are null capable. This value for FORMAT will only produce the desired results when the file is sent to another AS/400 with OS/400 release VRM420 and later installed.

Note: Specifying FORMAT(*F) may increase the amount of storage and time required when transmitting the network file. Specifying FORMAT(*U) also requires that the receiving system must have a database file created with the identical field mappings as the file being sent, in order to receive the file correctly.

Top

Send priority (PTY)

Specifies the queuing priority used for this file when it is being routed through a SNADS network.

The possible values are:

***NORMAL**
The file is sent with a service level priority of data low, which is used for most data traffic. On a System i5, data low distributions are placed on the normal distribution queue specified for the route.

***HIGH**
The file is sent with a service level priority of data high, which is used for high priority data traffic. On a System i5, data high distributions are placed on the data high distribution queue specified for the route.

Top

Examples

Example 1: Sending a Member

```
SNDNETF   TOUSRID((JONES SYSTEM1))  FILE(EMPLOYEE)  MBR(PGMR)
```

This command sends member PGMR of file EMPLOYEE to the user identified to the network with a user ID of (JONES SYSTEM1). The library list is used to locate the file.

Example 2: Sending a Nonsource File

```
SNDNETF   TOUSRID((JONES SYSTEM2))  FILE(EMPLOYEE)  MBR(PGMR)
          TOTYPE(*DATA)
```

This command sends member PGMR of file EMPLOYEE to the user identified to the network with a user ID of (JONES SYSTEM2). The library list is used to locate the file. The file is being sent as a nonsource file removing the sequence numbers and date fields.

Top

Error messages

***ESCAPE Messages**

CPF8057
File &1 in &2 not a physical file or save file.

CPF8058
File &1 is a spooled file.

CPF8059
Member name not allowed for save file.

CPF8063
Cannot assign necessary resource.

CPF8064
File &1 in &2 member &3 not sent to any users.

CPF8066
One or more user identifiers on this command is not correct.

CPF8068
Error detected while processing file to be sent.

CPF8072
Object to be sent is greater than maximum size of 2GB.

CPF9005
System resource required to complete this request not available.

CPF9006
User not enrolled in system distribution directory.

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.

CPF9812
File &1 in library &2 not found.

CPF9820
Not authorized to use library &1.

CPF9822
Not authorized to file &1 in library &2.

CPF9830
Cannot assign library &1.

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.

CPF9848
Cannot open file &1 in library &2 member &3.

CPF9849
Error while processing file &1 in library &2 member &3.

Top

Send Network Message (SNDNETMSG)

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

Parameters
Examples
Error messages

The Send Network Message (SNDNETMSG) command sends a message to another user on the local or a remote system through the SNADS network.

This message is sent as an informational message to the message queue that is defined for the recipient on the receiving system.

Note: The recipient must have a valid message queue specified in his user profile. Messages sent with the SNDNETMSG command are rejected if the recipient does not have a message queue specified in the user profile. The message queue specified in the network attributes is not used. Additional information on specifying a message queue when sending and receiving messages is in the CL topic collection in the Programming category in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/.

This command does not perform the CCSID translation of the messsage. However, it sends the job CCSID to the receiving system.The message is translated upon receipt if needed. The user ID and address of both the recipient and the originator are translated from the current job CCSID to the multinational character set 697/500.

Restriction: The user must be enrolled in the system distribution directory. A description of the system distribution directory is in the SNA Distribution Services book, SC41-5410.

Top

Parameters

Keyword	Description	Choices	Notes
MSG	Message text	*Character value*	Required, Positional 1
TOUSRID	User ID	Values (up to 50 repetitions): *Element list*	Required, Positional 2
	Element 1: User ID	*Character value*	
	Element 2: Address	*Character value*	

Top

Message text (MSG)

Specifies the message text of the immediate message that is sent. An immediate message is a message that is not stored in a message file. The text must be enclosed in apostrophes if it contains blanks or special characters. A maximum length of 256 characters can be specified.

This is a required parameter.

Top

User ID (TOUSRID)

Specifies one or more user IDs, or the name of one or more distribution lists containing user IDs of users to whom the message is to be sent.

A combination of both user IDs and distribution lists can be specified on the same command. Each user ID or distribution list is specified as a two-part name, and both parts are required. Up to 50 user IDs can be specified.

Note: Depending on the type of work station being used, the internal value for a user identifier may differ from the characters shown by the Display Directory Entries (DSPDIRE) command. If the byte-string value specified for the TOUSRID parameter does not match the rules for an internal user identifier value, or if it does not match the internal value for any enrolled user, an error may be reported.

This is a required parameter.

You can enter multiple values for this parameter.

Top

Examples

```
SNDNETMSG   MSG('I''m updating the accounts receivable files.')
            TOUSRID((SMITH SYSTEM2))
```

This command sends the message specified in the MSG parameter to the user identified to the network with a user ID of (SMITH SYSTEM2).

Top

Error messages

***ESCAPE Messages**

CPF8063
Cannot assign necessary resource.

CPF8066
One or more user identifiers on this command is not correct.

CPF8069
Message not sent to any users.

CPF9005
System resource required to complete this request not available.

CPF9006
User not enrolled in system distribution directory.

CPF9830
Cannot assign library &1.

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.

Top

Send Network Spooled File (SNDNETSPLF)

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

Parameters
Examples
Error messages

The Send Network Spooled File (SNDNETSPLF) command sends a spooled file to another user on the local system or on a remote system on the SNADS network. The file is placed on the output queue that is specified in the user profile of the user to whom the spooled file was sent.

When the file arrives at the destination system, a message is sent to both the recipient and sending user notifying them of the arrival of the spooled file.

This command does not perform any CCSID translation on the contents of the file. However, the user ID and address of both the recipient and the originator are translated from the current job CCSID to the multinational character set 697/500.

Restrictions:

1. The user must be enrolled in the system distribution directory to run this command. The sender must have read, add, and delete authority to the receiving output queue when sending to user on the same system.
2. One of the following must be true:
   - The requester is the creator of the file.
   - The requester has *READ authority to the output queue on which the file resides, and DSPDTA(*YES) was specified on the CRTOUTQ command.
   - The requester has *SPLCTL special authority.
   - The requester has *JOBCTL special authority, and the output queue on which the file resides has OPRCTL(*YES) specified on the CRTOUTQ command.
   - The output queue has DSPDTA(*YES) specified on the CRTOUTQ command.
   - The requester has owner authority to the output queue on which the file resides and the queue had AUTCHK(*OWNER) and DSPDTA(*YES) or DSPDTA(*NO) specified on the CRTOUTQ command.
   - The requester has *READ, *ADD, and *DELETE authority to the output queue on which the file resides and the queue has AUTCHK(*DTAAUT) and DSPDTA(*YES) or DSPDTA(*NO) specified on the CRTOUTQ command.
3. DTAFMT(*RCDDATA) must be used when sending a spooled file to a release prior to Version 1 Release 3 Modification 0 (V1R3).

Top

Parameters

Keyword	Description	Choices	Notes
FILE	Spooled file	*Name*	Required, Positional 1
TOUSRID	User ID	Values (up to 50 repetitions): *Element list*	Required, Positional 2
	Element 1: User ID	*Character value*	
	Element 2: Address	*Character value*	

Keyword	Description	Choices	Notes
JOB	Job name	Single values: * Other values: *Qualified job name*	Optional, Positional 3
	Qualifier 1: Job name	*Name*	
	Qualifier 2: User	*Name*	
	Qualifier 3: Number	000000-999999	
SPLNBR	Spooled file number	1-999999, ***ONLY**, *LAST, *ANY	Optional, Positional 4
JOBSYSNAME	Job system name	*Name*, ***ONLY**, *CURRENT, *ANY	Optional
CRTDATE	Spooled file created	Single values: ***ONLY**, *LAST Other values: *Element list*	Optional
	Element 1: Creation date	*Date*	
	Element 2: Creation time	*Time*, ***ONLY**, *LAST	
DTAFMT	Data format	***RCDDATA**, *ALLDATA	Optional
CLASS	VM/MVS class	**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	Optional
PTY	Send priority	***NORMAL**, *HIGH	Optional

Top

Spooled file (FILE)

Specifies the name of the spooled file that is sent to the specified user. The file name is the name of the device file that was used by the program to produce the spooled output file.

This is a required parameter.

Top

User ID (TOUSRID)

Specifies the two-part user ID of one or more users to whom the file is being sent, or the name of one or more distribution lists containing the user IDs of one or more users to whom the file is being sent. A combination of user IDs and distribution lists can be specified on the same command. Each user ID or distribution list is specified as a two-part name, and both parts are required. The users in the distribution list may be either remote or local.

Note: Depending on the type of work station being used, the internal value for a user identifier may differ from the characters shown by the DSPDIRE command. If the byte-string value specified for the TOUSRID parameter does not match the rules for an internal user identifier value, or if it does not match the internal value for any enrolled user, an error may be reported.

This is a required parameter.

You can enter multiple values for this parameter.

Top

Job name (JOB)

Specifies the name of the job that created the spooled output file whose data records are to be sent.

*****
The job that issued this command is the job that created the spooled file.

qualified-job-name
Specify the qualified name of the job that created the spooled file. If no job qualifier is given, all jobs currently in the system are searched for the simple job name.

Top

Spooled file number (SPLNBR)

Specifies the number of the spooled output file from the job whose data records are to be sent.

***ONLY**
Only one spooled file in the job has the specified file name; therefore, the number of the spooled file is not necessary.

***LAST**
The spooled file with the highest number and the specified file name is used.

***ANY**
The spooled file number is not used to determine which spooled file is used. Use this value when the job system name parameter or the spooled file create date and time parameter is to take precedence over the spooled file number when selecting a spooled file.

spooled-file-number
Specify the number of the spooled file having the specified file name whose data records are to be sent.

Top

Job system name (JOBSYSNAME)

Specifies the name of the system where the job that created the spooled file (JOB parameter) ran. This parameter is considered after the job name, user name, job number, spooled file name, and spooled file number parameter requirements have been met.

***ONLY**
There is one spooled file with the specified job name, user name, job number, spooled file name, spooled file number, and spooled file create date and time.

***CURRENT**
The spooled file created on the current system with the specified job name, user name, job number, spooled file name, spooled file number, and create date and time is used.

***ANY**
The job system name is not used to determine which spooled file is used. Use this value when the spooled file create date and time parameter is to take precedence over the job system name when selecting a spooled file.

name
Specify the name of the system where the job that created the spooled file ran.

Top

Spooled file created (CRTDATE)

Specifies the date and time the spooled file was created. This parameter is considered after the job name, user name, job number, spooled file name, spooled file number, and job system name parameter requirements have been met.

Single values

***ONLY**
There is one spooled file with the specified job name, user name, job number, spooled file name, spooled file number, and job system name.

***LAST**
The spooled file with the latest create date and time of the specified job name, user name, job number, spooled file name, spooled file number, and job system name is used.

Element 1: Creation date

date Specify the date the spooled file was created.

Element 2: Creation time

***ONLY**
There is one spooled file with the specified job name, user name, job number, spooled file name, spooled file number, job system name, and spooled file create date.

***LAST**
The spooled file with the latest create time of the specified job name, user name, job number, spooled file name, spooled file number, job system name, and spooled file create date is used.

time Specify the time the spooled file was created.

Top

Data format (DTAFMT)

Specifies the format in which to transmit the spooled file. This parameter is applicable only to spooled print files. It is ignored for diskette files.

Spooled files that contain special device requirements cannot be sent using this command if *RCDDATA is specified on the **Data format** prompt (DTAFMT parameter).

Note: The device requirements are listed as part of the attributes for the spooled file. To view the device requirements, use the WRKSPLFA command or the attributes option on the WRKSPLF command. If any of the device requirements are attributes of the file (if any of the device requirements on the display have a 'Y'), the spooled file cannot be sent. Either specify DTAFMT(*ALLDATA) or copy the spooled file to a database file using the Copy Spooled File (CPYSPLF) command and then use the Send Network File (SNDNETF) command to send the file as a data file.

If *RCDDATA is specified on the **Data format** prompt (DTAFMT parameter), the following attributes of the spooled file are kept:

- File name
- Number of copies
- Characters per inch
- Drawer
- Form type
- IGC data indicator
- Lines per inch
- Page length
- Page width
- Page rotation
- Font name
- Print text

- Diskette label
- Diskette creation and expiration dates
- Diskette code type
- Diskette exchange type

The possible values are:

***RCDDATA**
The spooled file is sent in the existing lower function format. Some functions will be removed from data that is sent in this format because the format does not support advanced functions. Use this format for sending spooled files to a System/36, System/38, or a System/370.

***ALLDATA**
The spooled file is sent as it exists, without loss of attributes. All spooled file attributes required to reproduce the file on the receiving system are also sent. Use the *ALLDATA value to send spooled files to Version 1, Release 3, Modification level 0 or later releases of the AS/400 system.

NOTES:

1. If a file will be printed on the receiving system, it must be printed on the same type of printer as it was intended to be printed on the source system.
2. Spool files that require advanced function printing (AFP) resources may print differently on the receiving system. To assure that spool files are printed in the same manner, the following must occur:
   - Before using this command to send a file to a user on the same system, make sure that the libraries containing non-IBM supplied AFP resources are in the library list.
   - Before using this command to send a file to a different user or system, make sure that the libraries containing non-IBM supplied AFP resources are in the initial library list of the user receiving the spooled files.
3. Use this format to send *LINE, *AFPDS, and *AFPDSLINE printer type device files to System 370.

Top

VM/MVS class (CLASS)

Specifies the VM/MVS SYSOUT class for distributions sent to a VM host system or to an MVS host system.

The possible values are:

A The class is A.

B-Z, 0-9
Specify the distribution class. Valid values range from B through Z and 0 through 9.

Top

Send priority (PTY)

Specifies the queuing priority used for this spooled file when it is being routed through a SNADS network.

The possible values are:

***NORMAL**
The spooled file is sent with a service level priority of data low, which is used for most data traffic. On a System i5, data low distributions are placed on the normal distribution queue specified for the route.

***HIGH**
The spooled file is sent with a service level priority of data high, which is used for high priority data traffic. On a System i5, data high distributions are placed on the data high distribution queue specified for the route.

Top

Examples

Example 1: Sending a Spooled File

```
SNDNETSPLF   FILE(QPRINT)  TOUSRID((JDE SYS1))
             JOB(142857/PAPER/PRINT)
             SPLNBR(*LAST)  DTAFMT(*ALLDATA)
```

This command sends the last (most recently created) copy of spooled file QPRINT from job 142857/PAPER/PRINT to the user with a user ID of JDE SYS1. All spooled file functions will be sent.

Example 2: Sending Print Attributes

```
SNDNETSPLF   DTAFMT(*RCDDATA)
```

This command sends a limited set of print attributes.

Example 3: Sending All Print Attributes

```
SNDNETSPLF   DTAFMT(*ALLDATA)
```

This command sends all print attributes. *ALLDATA is only valid when it is sent from one System i5 to another System i5.

Top

Error messages

***ESCAPE Messages**

CPF2207
Not authorized to use object &1 in library &3 type *&2.

CPF3205
File not created.

CPF3207
Member not added. Errors occurred.

CPF3303
File &1 not found in job &5/&4/&3.

CPF3309
No files named &1 are active.

CPF3330
Necessary resource not available.

CPF3340
More than one file with specified name found in job &5/&4/&3.

CPF3342
Job &5/&4/&3 not found.

CPF3343
Duplicate job names found.

CPF3344
File &1 number &8 no longer in the system.

CPF3429
File &1 number &7 cannot be displayed, copied, or sent.

CPF3482
Copy request failed. Spool file &1 is open.

CPF3486
CHLVAL parameter value not valid.

CPF3492
Not authorized to spooled file.

CPF3493
CTLCHAR parameter not correct for file &1.

CPF3499
Records in file &1 preceded all assigned channel values.

CPF8055
Spooled file contains special device requirements. File not sent.

CPF8063
Cannot assign necessary resource.

CPF8066
One or more user identifiers on this command is not correct.

CPF8067
File &1 not sent to any users.

CPF8068
Error detected while processing file to be sent.

CPF8072
Object to be sent is greater than maximum size of 2GB.

CPF9005
System resource required to complete this request not available.

CPF9006
User not enrolled in system distribution directory.

CPF9820
Not authorized to use library &1.

CPF9830
Cannot assign library &1.

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.

CPF9849
Error while processing file &1 in library &2 member &3.

Top

Send Program Message (SNDPGMMSG)

Where allowed to run: Compiled CL program or interpreted REXX (*BPGM *IPGM *BREXX *IREXX)
Threadsafe: Yes

Parameters
Examples
Error messages

The Send Program Message (SNDPGMMSG) command sends a message to a named message queue or to a call message queue. A call message queue can be the *EXT external message queue or a message queue associated with a call stack entry. Each time a program or procedure is called a new message queue is associated with its call stack entry. The message queue is identified by the name of its associated program or procedure.

A program can send a message to its own message queue or to a message queue that is associated with a different call stack entry.

This command can send both exception and non-exception messages.

Restrictions:

1. The SNDPGMMSG command allows a message of up to 512 characters to be sent. However, if the message is sent to the *EXT message queue of an interactive job, only 76 characters are shown on the Display Program Messages display. If the message is sent to a user's, work station's, or the system operator's message queue, the Display Message (DSPMSG) command allows all 512 characters to be displayed.
2. This command can only send inquiry messages (specified by MSGTYPE(*INQ)) to one message queue or to two nonprogram message queues if one of the queues is *HSTLOG.

Top

Parameters

Keyword	Description	Choices	Notes
MSG	Message text, or	*Character value*	Optional, Positional 1
MSGID	Message identifier	*Name*	Optional, Positional 2
MSGF	Message file	*Qualified object name*	Optional, Positional 3
	Qualifier 1: Message file	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
MSGDTA	Message data field values	*Character value*, ***NONE**	Optional, Positional 4
TOPGMQ	Call stack entry message queue	Single values: *EXT Other values: *Element list*	Optional
	Element 1: Relationship	***PRV**, *SAME	
	Element 2: Call stack entry identifier	*Element list*	
	Element 1: Call stack entry	*Character value*, *	
	Element 2: Module	*Name*, ***NONE**	
	Element 3: Bound program	*Name*, ***NONE**	

Keyword	Description	Choices	Notes
TOMSGQ	Send to non-pgm message queue	Single values: ***TOPGMQ**, *SYSOPR Other values (up to 50 repetitions): *Qualified object name*	Optional
	Qualifier 1: Send to non-pgm message queue	*Name*, *HSTLOG	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
TOUSR	To user profile	*Name*, *SYSOPR, *ALLACT, *REQUESTER	Optional
MSGTYPE	Message type	***INFO**, *INQ, *RQS, *COMP, *DIAG, *NOTIFY, *ESCAPE, *STATUS	Optional
RPYMSGQ	Message queue to get reply	Single values: ***PGMQ** Other values: *Qualified object name*	Optional
	Qualifier 1: Message queue to get reply	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
KEYVAR	CL var for KEYVAR (4)	*Character value*	Optional
CCSID	Coded character set ID	1-65535, *HEX, ***JOB**	Optional

Top

Message text, or (MSG)

Specifies the message text that is to be sent. A maximum of 3000 characters can be specified or, if you are prompting for this command in an interactive job, a maximum of 512 characters can be specified. The string must be enclosed in apostrophes if special characters (including blanks) are used. If this parameter is specified, a value cannot be specified for the **Message identifier (MSGID)** parameter, and *ESCAPE, *NOTIFY, or *STATUS cannot be specified for the **Message type (MSGTYPE)** parameter. If this parameter is specified, a value cannot be specified for the **Message file (MSGF)** parameter or the **Message data field values (MSGDTA)** parameter, because these types require that a message identifier also be specified.

Coded Character Set Identifier (CCSID) Considerations

The text supplied for the MSG parameter is assumed to be in the CCSID of the job running this command unless a coded character set identifier is supplied in the CCSID parameter. For more information about the message handler and its use of CCSIDs, see the i5/OS globalization topic collection in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/.

Top

Message identifier (MSGID)

Specifies the message identifier of a message description whose predefined message is being sent by the program to a message queue. If this parameter is specified, a value cannot be specified for the **Message text, or (MSG)** parameter.

Top

Message file (MSGF)

Specifies the message file that contains the predefined message to be sent. This parameter is required if a value is specified for the **Message identifier (MSGID)** parameter.

Qualifier 1: Message file

name Specify the name of the message file which contains the predefined message to be 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 job is used to locate the message file. If no current library entry exists in the library list, QGPL is used.

name Specify the library where the message file is located.

Top

Message data field values (MSGDTA)

Specifies the character string, or a CL variable that contains a character string, containing one or more substitution values that are used as message data fields within the predefined message. The substitution values take the place of the substitution variables that were defined in the message text when the message was defined.

***NONE**
There are no program-supplied substitution values used in the specified message.

character-string
Specify the character string that gives the substitution values in the specified predefined message that is sent by the program, or specify the name of the CL variable that contains the character string.

Coded Character Set Identifier (CCSID) Considerations

The text supplied for the MSGDTA parameter that corresponds to the *CCHAR type field is assumed to be in the CCSID of the job running this command unless the coded character set identifier is supplied in the CCSID parameter. All other text supplied for the MSGDTA parameter is assumed to be 65535 and is not converted. For more information about the message handler and its use of CCSIDs, see the i5/OS globalization topic collection in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/. For more information about the *CCHAR type fields, see the Add Message Description (ADDMSGD) command.

Top

Call stack entry message queue (TOPGMQ)

Specifies the call message queue to which the specified message is to be sent. The message queue can be the *EXT external queue or the call message queue associated with a call stack entry.

Single values

***EXT** The message is sent to the external message queue of the job. The external message queue is used to communicate with the external requester of the job, such as a display station user. *INQ messages that are sent to *EXT wait for 24 hours before the default reply is sent.

Messages sent to this queue can be 512 characters in length, but only 76 characters of text are shown on the Program Messages display.

Element 1: Relationship

Two parameter elements are used to specify the call stack entry message queue from which a message is to be removed. The first element specifies whether the message queue is associated with the program or procedure identified by the second element, or if it is associated with the caller of the program or procedure.

***PRV** The message is sent to the message queue of the call stack entry that is immediately previous to the one identified by the second element of this parameter. However, if the message queue immediately previous to the one identified by the second element is for an Integrated Language Environment (ILE) program entry procedure (PEP), the message is sent to the message queue that precedes the PEP message queue in the stack.

***SAME**
The message is sent to the message queue of the call stack entry identified by the second element of this parameter.

Element 2: Call stack entry identifier

The second element of this parameter has three elements. Element 1 specifies an OPM program or ILE procedure name or a special value. Element 2 specifies an ILE module name which is used as a qualifier for the value specified in element 1. Element 3 can specify either an OPM program name or an ILE program name or a service program name, depending on what is specified in element 1. Element 3 is also used as a qualifier for what is specified in element 1.

Element 1: Call stack entry

***** Specifies the OPM program or ILE procedure running this command.

name Specify the name of the OPM program or ILE procedure used to identify the call stack entry.

If this element identifies an OPM program, the name specified can be a maximum of 10 characters. If this element identifies an ILE procedure, the name specified can be a maximum of 256 characters.

Nested procedure names can be specified by separating each procedure name with a colon (:). When specifying nested procedure names, the outermost procedure name is identified first, followed by its contained procedures. The innermost procedure name is identified last in the string.

Partial names of programs or procedures can be specified by placing three less-than symbols (<<<) at the beginning of the name or by placing three greater-than symbols (>>>) at the end of the name. If both the greater-than symbols and the less-than symbols are used, the program or procedure name specified is limited to 250 characters.

The system begins its search for the specified program or procedure name with the most recently called program or procedure.

When searching for a partial program or procedure name:

- The less-than symbols (<<<) are truncated when specified only at the beginning of a program or procedure name and the remaining character string is right-justified. The remaining characters in the specified string are compared to the current program or procedure on the call stack, starting with the last position of the program or procedure name and comparing backward.

- The greater-than symbols (>>>) are truncated when specified only at the end of a program or procedure name. The remaining characters in the specified string are compared to the current program or procedure on the call stack, starting with the first position of the program or procedure name.
- The less-than symbols (<<<) and the greater-than symbols (>>>) are truncated when both are specified for a program or procedure name. The remaining characters are used to scan and compare the entire length of the specified string with the current program or procedure on the call stack.

Element 2: Module

***NONE**
No ILE module qualifier is provided.

name Specify the ILE module name to be used to identify the message queue.

Element 3: Program

***NONE**
No program qualifier is provided.

name Specify the program name to be used to identify the message queue.

The procedure name alone may not identify the correct procedure. Several different procedures with the same name can run in a job. To further identify a procedure, the name specified can be qualified by a module name, or by both a module name and a bound program name. The following special values can be specified for the first qualifier of the second element of this parameter:

***CTLBDY**
Specifies the call stack entry that is at the most recent control boundary. This entry will be running in the same activation group as the CL program that is running the SNDPGMMSG command. Note that a control boundary will not exist if all programs on the call stack are OPM programs.

***PGMBDY**
Specifies the program boundary of either the program that is using the SNDPGMMSG command or the program whose name is specified for qualifier 3 of this parameter. If no name is specified for qualifier 3, it is assumed that the program is the one using the command.

If it is an ILE program that is being specified, this special value identifies the call stack entry for the program entry procedure (PEP) of that program, if the program was called by a dynamic call. If the program was called by a procedure pointer, this special value identifies the call stack entry for the procedure that was pointed to. If it is an ILE service program that is being specified, this special value identifies the call stack entry for the first procedure that was called in that service program.

If the program being specified is an OPM program, this special value has the same effect as specifying the special value * or a program name for item 1. A difference will occur if the OPM program has called itself recursively. In this case, this special value identifies the first recursion level rather than the current recursion level as would be the case if the special value * or a program name was used.

***PGMNAME**
Specifies that the call stack entry will be identified only by using a program name and optionally a module name. When this special value is used, qualifier 3 must specify an ILE program or service program name or OPM program name. Qualifier 2 may contain either the special value *NONE or an ILE module name.

This special value is used to send a message to the most recently called procedure that is part of the specified ILE program or service program. When using this special value, it is not necessary to explicitly provide a procedure name. If a module name is also provided, then this special value is used to send a message to the most recently called procedure that is both part of the identified program and the identified module.

This special value may also be used to send a message to an OPM program. In this case, using this special value and providing the OPM program name in item 3 has exactly the same effect as providing that program name here in item 1. Note that if this special value is being used to send to an OPM program then the module name must be specified as *NONE.

Top

Send to non-pgm message queue (TOMSGQ)

Specifies up to 50 nonprogram message queues to which an informational message is sent. For an inquiry message, one message queue may be specified or two message queues may be specified if one of the queues is *HSTLOG. This parameter cannot be used if a value is specified for the **To user profile (TOUSR)** parameter.

Single values

***TOPGMQ**
The message is sent only to the call message queue specified for the **Call stack entry message queue (TOPGMQ)** parameter.

***SYSOPR**
The message is sent to the system operator message (message queue QSYSOPR in library QSYS). Any message sent to message queue QSYSOPR in library QSYS automatically has a copy of the message sent to the QHST (history log) message queue in library QSYS.

Qualifier 1: Send to non-pgm message queue

***HSTLOG**
The message is sent to the system history log (message queue QHST in library QSYS). If *HSTLOG is specified more than once, only one message will be sent to the system history log. If *HSTLOG is specified with message queue QSYSOPR, only one message is sent to the system history log.

name Specify the name of the message queue to which the message is to be sent. A maximum of fifty message queues can be specified.

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 message queue. If no current library entry exists in the library list, QGPL is used.

name Specify the library where the message queue is located.

Top

To user profile (TOUSR)

Specifies that the message is to be sent to the message queue specified in the user profile for the user named on this parameter. This parameter cannot be used if a value is specified for the **Send to non-pgm message queue (TOMSGQ)** parameter or the **Call stack entry message queue (TOPGMQ)** parameter.

***SYSOPR**
The message is sent to the system operator (message queue QSYSOPR in library QSYS). Any message sent to message queue QSYSOPR in library QSYS automatically has a copy of the message sent to the QHST (history log) message queue in library QSYS.

***REQUESTER**
The message is sent to the user profile message queue for interactive jobs or to the system operator's message queue (QSYSOPR in library QSYS) for batch jobs.

***ALLACT**
A copy of the message is sent to the user profile message queue of each user profile with an interactive job currently running. *ALLACT cannot be specified with inquiry messages.

name Specify the user profile name of the user to whom the message is to be sent.

Top

Message type (MSGTYPE)

Specifies which message type is assigned to this message when it is sent by this program.

Notes:

1. Inquiry messages can be sent only to the external queue or to a named message queue specified for the TOUSR or TOMSGQ parameters. When sending an inquiry with the TOMSGQ parameter, a second queue can be specified if the value is *HSTLOG.
2. Completion, diagnostic, escape, notify, and status messages can be sent only to a call message queue.
3. Escape messages cannot be sent to the external message queue.

***INFO**
The message is sent as an informational message.

***INQ** The message is sent as an inquiry message.

***COMP**
A completion message is sent to a call message queue. A completion message indicates the status of the work that is successfully performed.

***DIAG**
A diagnostic message is sent to a call message queue. Diagnostic messages provide information about errors detected by this program. The errors are either in the input sent to it, or are those that occurred while it was running the requested function. An escape or notify message should also be sent to inform the receiving program or procedure of the diagnostic messages that are on its message queue.

***NOTIFY**
A notify exception message is sent to a call message queue. A notify message describes a condition for which corrective action must be taken before the sending program can continue. A reply message is sent back to the sending program. After corrective action is taken, the sending program can resume running and can receive the reply message from its message queue.

***ESCAPE**
An escape exception message is sent to a call message queue. An escape message describes an irrecoverable error condition. The sending program does not continue to run.

***RQS** A request message is sent to a call message queue. A request message allows request data received from device files to pass from this program to another program or procedure. An immediate message, specified by the MSG parameter, must be used to send the request.

***STATUS**
A status exception message is sent to a call message queue. The status message describes the

status of work performed by the sending program. The first 28 characters of message data in the MSGDTA parameter are used as the comparison data for message monitors (established by the Monitor Message (MONMSG) command). If the status exception message is not being monitored, control is returned to the sending program. If a status message is sent to the external message queue of an interactive job, the message is shown on line 24, processing continues, and no response is required.

Note: This value cannot be specified if the **Message text, or (MSG)** parameter, is specified.

Top

Message queue to get reply (RPYMSGQ)

Specifies, for inquiry and notify messages only, the call message queue or the non-program message queue to which the reply message is to be sent.

Single values

***PGMQ**
The reply to an inquiry or notify message is sent to the message queue associated with the call stack entry of the program or procedure using this command.

Qualifier 1: Message queue to get reply

name Specify the name of the message queue to which the reply is 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 job is used to locate the message queue. If no current library entry exists in the library list, QGPL is used.

name Specify the library where the message queue is located.

Top

CL var for KEYVAR (4) (KEYVAR)

Specifies the name of the CL character variable, if any, that contains the message reference key that identifies the message sent by the program containing this command. The message reference key is assigned by the system when the message is sent and is placed in the variable specified here.

If a message is being sent to a message queue associated with a call stack entry, KEYVAR refers to that message queue (specified for the **Call stack entry message queue (TOPGMQ)** parameter). If *INQ or *NOTIFY is specified for the **Message type (MSGTYPE)** parameter, KEYVAR refers to the message queue specified for the **Message queue to get reply (RPYMSGQ)** parameter. In all other cases, KEYVAR refers to the message queue specified for the TOPGMQ parameter.

Any type of message can be assigned a key when it is being sent to a program message queue. For messages sent to a nonprogram message queue, message reference keys are available for inquiry (*INQ) messages only. If another message type is sent to a nonprogram queue, no message key is available and blanks are returned for KEYVAR.

The variable must be a character variable having a length of 4 characters. If KEYVAR is not specified and a reply is required, it can be received by the program in FIFO order.

Top

Coded character set ID (CCSID)

Specifies the coded character set identifier (CCSID) that the supplied message or message data is in. If a message identifier is specified, the text supplied by the MSGDTA (message data) parameter that corresponds to the *CCHAR type field is assumed to be in the CCSID specified by the CCSID parameter. The data supplied that does not correspond to the *CCHAR type field is assumed to be 65535 and is not converted. For more information about the *CCHAR type field see the Add Message Description (ADDMSGD) command.

If no message identifier is specified, the text supplied by the MSG (message) parameter is assumed to be in the CCSID supplied by the CCSID parameter. For more information about the message handler and its use of CCSIDs, see the i5/OS globalization topic collection in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/.

***JOB** The message data or immediate text is assumed to be in the CCSID of the job running this command.

***HEX** The message data or immediate text is not converted. CCSID 65535 is used.

coded-character-set-identifier
Specify a valid CCSID in which you want your message or message data to be considered in. Valid values range from 1 through 65535. This command validates the CCSID. See the Globalization information in the iSeries Information Center at http://www.ibm.com/eserver/iseries/infocenter for a list of valid CCSID values.

Top

Examples

Example 1: Specifying Substitution Values

```
SNDPGMMSG   MSGID(UIN0023)  MSGF(INV)  MSGDTA('50 100')
            TOPGMQ(*EXT)
```

This command sends the message identified as UIN0023, which is stored in message file INV, to the external message queue of the job (the Display Program Messages presents the message at a display station). The data, which contains two substitution values specified in the MSGDTA parameter, is sent with the message. This data can then be used as substitution values when the message is received, or it can be used as data to be dumped, depending on how the message UIN0023 is defined in the message file. Assuming that the variables &1 and &2 have been defined in the message file as character variables, each 3 characters long, and that the first-level message text of the message UIN0023 is: ’Requested item decreased by &1; current balance &2.’ The message text sent is: ’Requested item decreased by 50; current balance 100.’

Example 2: Sending an Inquiry Message

```
SNDPGMMSG   MSG('Mount checks in printer before continuing')
            MSGTYPE(*INQ)  TOMSGQ(*SYSOPR)
```

This command sends an inquiry message to the system operator. The operator looks at the message that was sent by using the DSPMSG command and responds to the message directly on that display. A Receive Message (RCVMSG) command is used in the program to accept the operator’s response.

Example 3: Sending an Escape Message

```
SNDPGMMSG   MSGID(USR0001)  MSGF(USRMSGR)  TOPGMQ(*PRV  *)
            MSGTYPE(*ESCAPE)
```

This command is an example of how a message could be sent to the caller of a program or procedure to cause an abnormal end. The message USR0001 could indicate that an invalid code was passed (such as a nonnumeric code when numeric is required). Because the message being sent is an escape message, the program or procedure that is sending the message cannot be resumed. The values *PRV and * did not have to be coded on this command because they are the default values on the TOPGMQ parameter.

Example 4: Sending an Escape Message to an ILE Procedure

```
SNDPGMMSG   MSGID(USR0001)  MSGF(USRMSGR)
            TOPGMQ(*SAME ACCOUNT_FINAL_TOTALS)
            MSGTYPE(*ESCAPE)
```

This command sends a message to an ILE procedure. In this example, the call stack entry identifier is more than 10 characters. Since no qualifier is specified, the actual module name and bound program name associated with the procedure are not used in finding the procedure. The escape exception message is sent to the message queue associated with ACCOUNT_FINAL_TOTALS because *SAME is specified for Element 1.

Example 5: Sending an Escape Message using Qualifiers

```
SNDPGMMSG   MSGID(USR0001)  MSGF(USRMSGR)
            TOPGMQ(*PRV FIRST_QTR_SUMMARY SUMQTRS REPORTS)
            MSGTYPE(*ESCAPE)
```

This command sends an escape exception message to the caller of the procedure FIRST_QTR_SYMMARY. The procedure is qualified by the module name SUMQTRS and the bound program name REPORTS. The escape exception message interrupts the sending program and the sending program is not resumed.

Example 6: Sending a Completion Message using a Partial Procedure Name

```
SNDPGMMSG   MSGID(USR0001)  MSGF(USRMSGR)
            TOPGMQ(*SAME 'MANAGE_SALES>>>')  MSGTYPE(*COMP)
```

This command sends a completion message to the most recent procedure whose name begins with MANAGE_SALES.

Top

Error messages

***ESCAPE Messages**

CPF24CB
*PGMNAME requires a specified program name.

CPF2409
Specified message type not valid with specified program message queue.

CPF2428
Message queue parameter is not valid.

CPF2453
Reply queue not sender’s program message queue.

CPF2469
Error occurred when sending message&1.

CPF247A
Call stack entry not found.

CPF247E
CCSID &1 is not valid.

CPF2499
Message identifier &1 not valid.

CPF2524
Exception handler not available because of reason code &1.

CPF2550
Exception message sent to a deleted program or procedure.

CPF2702
Device description &1 not found.

CPF7C08
No support network connection.

CPF8C0C
Content of problem record &1 not valid.

CPF8C0E
Library QGPL not found.

CPF8C01
Cannot connect to IBM service system. One session allowed.

CPF8C07
A parameter is not valid.

CPF8C08
Cannot specify *SELECT for the control point name.

CPF8C09
&1 not defined as a service provider.

CPF8C16
Error occurred while processing request.

CPF8C17
Sign-on failed.

CPF8C18
No support network connection.

CPF8C19
Remote support application failed.

CPF8C2A
Cannot connect to IBM service system.

CPF8C24
Error occurred while processing request.

CPF8C27
Alternate load device not found.

CPF8C32
PTF order cannot be processed.

CPF9830
Cannot assign library &1.

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.

Top

Send PTF Order (SNDPTFORD)

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

Parameters
Examples
Error messages

The Send Program Temporary Fix Order (SNDPTFORD) command allows you to prepare an order for:

- Individual PTFs
- Cumulative PTF package
- PTF Group
- Summary information for available PTFs
- Preventive Service Planning (PSP) information

Restrictions:

1. This command is shipped with public *EXCLUDE authority, and the QSRV and QSRVBAS user profiles have private authority to use the command.

2. The following restrictions apply for the IMGDIR parameter:
- You must have *X authority to each directory in the path.
- You must have *WX authority to the directory that contains optical image.

3. The following restrictions apply for the IMGCLG parameter:
- You must have the authorities needed to run the Create Image Catalog (CRTIMGCLG) command.
- You must have read(*READ) and add(*ADD) authorities to the QUSRSYS library.

Top

Parameters

Keyword	Description	Choices	Notes
PTFID	PTF description	Values (up to 20 repetitions): *Element list*	Required, Positional 1
	Element 1: PTF identifier	*Character value*, *CUMPKG, *HIPERGRP, *DB2GRP, *BRSGRP, *JVAGRP, *HTTPGRP, *PFRGRP	
	Element 2: Product	*Character value*, ***ONLYPRD**	
	Element 3: Release	*Character value*, ***ONLYRLS**	
PTFPART	PTF parts	***ALL**, *CVRLTR	Optional
RMTCPNAME	Remote control point	*Communications name*, ***IBMSRV**, *SELECT	Optional
RMTNETID	Remote network identifier	*Communications name*, ***NETATR**	Optional
DELIVERY	Delivery method	***LINKONLY**, *ANY	Optional
DLVRYFMT	Delivery format	***SAVF**, *IMAGE	Optional
ORDER	Order	***REQUIRED**, *PTFID	Optional
REORDER	Reorder	***NO**, *YES	Optional
CHKPTF	Check PTF	***NO**, *YES	Optional
IMGOPT	Image option	***DOWNLOAD**, *ORDER	Optional
IMGDIR	Image directory	*Path name*, ***DFT**	Optional
IMGPFX	Image prefix	*Character value*, ***SRVPVD**	Optional

Keyword	Description	Choices	Notes
IMGCLG	Image catalog	*Name*, ***NONE**	Optional

Top

PTF identifier (PTFID)

Specifies the list of PTFs being ordered.

Element 1: PTF identifier

***CUMPKG**
Order the latest level of the cumulative PTF package (SF99vrm) for the operating system release that is installed on the system. HIPER and DB2 Universal Database (UDB) group PTFs are automatically requested with this order. It cannot be ordered with any other PTF identifier or special value.

***HIPERGRP**
Order the latest level of HIPER PTF group for the operating system release that is installed on the system.

***DB2GRP**
Order the latest level of DB2 UDB PTF group for the operating system release that is installed on the system.

***BRSGRP**
Order the latest level of Backup Recovery Solutions PTF group for the operating system release that is installed on the system.

***HTTPGRP**
Order the latest level of IBM HTTP Server PTF group for the operating system release that is installed on the system.

***JVAGRP**
Order the latest level of Java PTF group for the operating system release that is installed on the system.

***PFRGRP**
Order the latest level of Performance Tools PTF group for the operating system release that is installed on the system.

character-value
Specify the PTF identifier. Some PTFs must be ordered individually or within a list of PTFs with the same prefix and not as part of a general list.

A cumulative PTF package is specified using the format SF99vrm and the Preventive Service Planning (PSP) format is SF98vrm, where vrm is version-release-modification.

Element 2: Product

***ONLYPRD**
The PTF identifier is associated with only one product.

Note: If this value and ORDER(*REQUIRED) are specified, requisites are sent for only that product that is installed or supported on your system.

character-value
Specify the 7-character product identifier of the product that the PTF is associated with. If the PTF identifier is associated with more than one product, the PTF order is limited to the product specified.

Element 3: Release

***ONLYRLS**
The PTF identifier is associated with only one release.

Note: If this value and ORDER(*REQUIRED) are specified, requisites are sent for only that release level that is installed or supported on your system.

character-value
Specify the 6-character release level in VxRyMz format where Vx is the version number, Ry is the release number, and Mz is the modification level. The variables x and y can be a number from 0 through 9, and the variable z can be a number from 0 through 9 or a letter from A through Z.

Top

PTF parts (PTFPART)

Specifies whether PTFs or cover letters are being ordered.

***ALL** PTFs and cover letters are being ordered.

***CVRLTR**
Cover letters only are being ordered.

Top

Remote control point (RMTCPNAME)

Specifies the remote control point of the service provider to whom the service request is sent.

***IBMSRV**
The service request is sent to IBM service support.

***SELECT**
A list of service providers is shown so the user can select the destination for the service request.

communications-name
Specify the name of the remote control point.

Top

Remote network identifier (RMTNETID)

Specifies the remote name identifier of the service provider's network.

***NETATR**
The service provider is in the local network.

communications-name
Specify the network name of the service provider to whom the request is being sent.

Top

Delivery method (DELIVERY)

Specifies how the PTFs are delivered.

***LINKONLY**
PTFs are delivered by the electronic customer support service link only. If the PTFs are too large to use service link, the command will fail.

***ANY** PTFs are delivered by any available method. The service link is used for most PTFs. PTFs that are too large for the service link are sent on the selected medium.

Top

Delivery format (DLVRYFMT)

Specifies the format of the delivered PTFs.

***SAVF** PTFs are delivered through the electronic customer support service link as save files.

***IMAGE**
PTFs are delivered through the electronic customer support service link as optical image files. Optical image files will contain PTFs and cover letters. The optical image file will be stored in the directory specified by the IMGDIR parameter.

Top

Order (ORDER)

Specifies the level of fixes that are being requested.

***REQUIRED**
The PTF ordered and its requisites are being requested.

***PTFID**
The specific PTF ordered is the one being requested. No requisites are sent.

Top

Reorder (REORDER)

Specifies whether a PTF that is currently loaded, applied, or on order should be ordered again.

***NO** PTFs that are already loaded, applied, or on order are not reordered.

***YES** PTFs that are already loaded, applied, or on order are reordered.

Note: A PTF is not reordered if the *SAVF delivery format is specified and a save file is available on the system.

Top

Check PTF (CHKPTF)

Specifies whether checking is performed on the service requester system to determine if PTFs are ordered based on whether or not the PTF product is installed or supported.

***NO** The PTFs specified on the PTF identifier (PTFID) parameter are ordered even when the PTF product is not installed or supported on the service requester.

***YES** The PTFs specified on the PTF identifier (PTFID) parameter are ordered only if the PTF product is installed or supported on the service requester.

Top

Image option (IMGOPT)

Specifies the option of the image order.

***DOWNLOAD**
The optical image can be ordered and downloaded automatically when the optical image is ready to download.

***ORDER**
The optical image will be ordered but it will not be automatically downloaded. The Work with PTF Order (WRKPTFORD) command can be used to display PTF orders information, download or cancel the order.

Note: The user has up to 14 days to resume or cancel a PTF order because the Fix Service Provider will close all orders after 14 days.

Top

Image directory (IMGDIR)

Specifies the directory where the optical image files are stored. If IMGCLG parameter is specified, the directory specified will be associated with the image catalog.

***DFT** The optical image files are stored in /QIBM/UserData/OS/Service/ECS directory.

path-name
:* @A1C From: Specify an existing directory. For more information on specifying path names, refer to ″Object naming rules″ in the ″CL concepts and reference″ topic in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/.

Top

Image prefix (IMGPFX)

Specifies a prefix for the optical image file names.

***SRVPVD**
The files will be named by the Service Provider.

character-value
Specify the prefix that will be used when naming the optical images being stored. If multiple images are received under one order, the files will be uniquely identified by a numerical suffix on the image name. For more information on specifying file names, refer to ″Object naming rules″ in the ″CL concepts and reference″ topic in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/.

Top

Image catalog (IMGCLG)

Specifies the image catalog to create based on the PTFs ordered.

***NONE**
No image catalog will be created.

name Specify the name that will be used when creating the image catalog. If the image catalog already exists, it will be used to store the images download.

Top

Examples

Example 1: Sending Order for PTFs by PTF Number Only

```
SNDPTFORD   PTFID((SI12345) (SI12346))
```

This command sends a request for PTF numbers SI12345 and SI12346.

Example 2: Sending Order for PTF by Product and Release

```
SNDPTFORD   PTFID((SI12345 5722SS1 V5R3M0))
            DELIVERY(*ANY)  ORDER(*REQUIRED)
```

This command sends a request for PTF SI12345 for Version 5 Release 3 Modification 0 of the operating system product (5722SS1). The PTF can be delivered by any available method. Any requisites are sent with the PTFs.

Example 3: Sending Order for Latest Cumulative PTF

```
SNDPTFORD   PTFID((*CUMPKG))
```

This command requests that the latest PTF cumulative package be sent for the operating system release level that is installed on your system.

Example 4: Sending Order for a PTF Group

```
SNDPTFORD   PTFID((SF99893))
```

This command sends a request for PTF group number SF99893.

Example 5: Sending Order for a DB2 UDB Group with *IMAGE Delivery Format

```
SNDPTFORD   PTFID((*DB2GRP))  DELIVERY(*LINKONLY)
            DLVRYFMT(*IMAGE)
```

This command sends a request for the latest level of the DB2 UDB PTF group for the operating system release that is installed on your system. The PTF group will be stored in optical image files inside /QIBM/UserData/OS/Service/ECS directory.

Example 6: Sending Order with *IMAGE Delivery Format

```
SNDPTFORD   PTFID((*DB2GRP))  DELIVERY(*LINKONLY)
            DLVRYFMT(*IMAGE)  IMGOPT(*ORDER)
```

This command sends a request for the latest level of the DB2 UDB PTF group for the operating system release that is installed on your system. The PTF order will be ordered but it will not be donwloaded. The status of this PTF order will be ″onorder″ and WRKPTFORD command can be used to display PTF order information, download or cancel the order.

Example 7: Sending Order with Image Catalog Option

```
SNDPTFORD   PTFID((*HIPERGRP)) DELIVERY(*LINKONLY)
            DLVRYFMT(*IMAGE) IMGCLG('MYCATALOG')
```

This command sends a request for the latest level of the Hiper PTF group for the operating system release that is installed on your system. The PTF image will be stored in the MYCATALOG image catalog.

Top

Error messages

***ESCAPE Messages**

CPF2702
Device description &1 not found.

CPF7C08
No support network connection.

CPF8C0C
Content of problem record &1 not valid.

CPF8C0E
Library QGPL not found.

CPF8C01
Cannot connect to IBM service system. One session allowed.

CPF8C07
A parameter is not valid.

CPF8C08
Cannot specify *SELECT for the control point name.

CPF8C09
&1 not defined as a service provider.

CPF8C16
Error occurred while processing request.

CPF8C17
Sign-on failed.

CPF8C18
No support network connection.

CPF8C19
Remote support application failed.

CPF8C2A
Cannot connect to IBM service system.

CPF8C24
Error occurred while processing request.

CPF8C27
Alternate load device not found.

CPF8C32
PTF order cannot be processed.

CPF8C99
PTF &2-&1 &3 not ordered.

CPF9846
Error while processing file &1 in library &2.

Top

Send/Receive File (SNDRCVF)

Where allowed to run:

- Batch program (*BPGM)
- Interactive program (*IPGM)

Threadsafe: No

Parameters
Examples
Error messages

The Send/Receive File (SNDRCVF) command is used by a CL program or ILE CL procedure to send data to and receive data from a device that is being used interactively by a user. The data is passed between the program in which the SNDRCVF command is used and the display device identified in the command. The data is passed using the display device file that was declared in the program. (A Declare File (DCLF) command included in the source used to compile the program was used to declare the file.) The data for each send/receive operation is passed as one record in a format identified by the RCDFMT parameter of this command (the format is defined in the data description specifications (DDS)). One CL variable is used for each field of the record format to pass the data. The CL variables used (including DDS indicators) are declared implicitly.

Of the record formats specified in the DCLF command, only one can be specified in each SNDRCVF command. If the device file has not been opened, it is opened by this command. The file and record format specified in this command can be overridden by an Override with Display File (OVRDSPF) command if that command is entered before the file is opened. However, care should be taken that the fields in the overriding record format correspond to the CL variables declared in the program.

Restrictions: This command is valid only within a CL program or ILE CL procedure and only for display files. It cannot be used with database files.

Top

Parameters

Keyword	Description	Choices	Notes
DEV	Display device	*Name*, ***FILE**	Optional, Positional 1
RCDFMT	Record format	*Name*, ***FILE**	Optional, Positional 2
OPNID	Open file identifier	*Simple name*, ***NONE**	Optional
WAIT	Wait	***YES**, *NO	Optional

Top

Display device (DEV)

Specifies the name of the display device that the data is to be sent to and the user's data is to be received from. A CL variable can be specified for this parameter so that the device name can be changed without changing the command.

***FILE** The data is to be sent to and received from the device associated with the device file (the device file that was declared in the FILE parameter of the DCLF command). If more than one device name is specified in the device file, *FILE cannot be specified.

name Specify the name of the device or the name of the CL variable that contains the name of the device that the CL program or ILE CL procedure is to send data to and receive data from. If a CL variable name is used in this parameter, only one SNDRCVF command is needed in the program to receive data from several devices.

Top

Record format (RCDFMT)

Specifies the name of the record format that is to be used to pass the data between the CL program or ILE CL procedure and the user. The format contains all the fields in the record. This parameter must be coded with a record format name if there is more than one record format in the device file; *FILE cannot be coded if there is more than one. SNDRCVF ignores the INVITE DDS keyword.

***FILE** There is only one record format in the device file; that format is to be used to send the data to and receive the data from the user.

name Specifies the name of the record format in which the data is to be sent to and received from the user. A CL variable name cannot be used to specify the record format name.

Top

Open file identifier (OPNID)

Specifies the open file identifier that was declared on a preceding Declare File (DCLF) command in the same CL program or ILE CL procedure. A CL variable cannot be specified for this parameter value.

***NONE**
No open file identifier is provided. This command will use the file associated with the DCLF command that had *NONE specified for the OPNID parameter. Only one file can be declared in a CL program or ILE CL procedure with *NONE as the open file identifier.

simple-name
Specify a name that matches the OPNID parameter value on a preceding DCLF command in the same CL program or ILE CL procedure.

Top

Wait (WAIT)

Specifies whether the CL program or ILE CL procedure either waits to receive the data from the user's device or continues to process the commands that follow this SNDRCVF command. If WAIT(*NO) is specified, the program must issue a WAIT command later in the program to complete the input operation.

Note: A CL variable cannot be coded on this parameter.

***YES** The program waits until the input operation from the device is completed; the next command is not processed until then.

***NO** The program does not wait for the input data; it continues to process commands until a WAIT command is reached later in the program.

Top

Examples

Example 1: Using Display File with One Record Format

```
DCLF   FILE(MENU1)
 :
SNDRCVF
```

This command sends and receives user data by way of the device file MENU1. Only one record format exists in the file. The device used is specified in the file.

Example 2: Using Display File with Multiple Record Formats

```
DCLF   FILE(SCR)  RCDFMT(REC8)
 :
SNDRCVF   RCDFMT(REC8)
```

The CL program or ILE CL procedure sends data to a user and receives data for the user who is using the device named in the device file (*FILE is assumed because DEV is not specified). The data is passed in the format specified by REC8 record format in the device file named SCR. The CL program or ILE CL procedure waits for the user data before continuing.

Example 3: Using a CL Variable for Device Name

```
DCLF   FILE(DF1)  RCDFMT(REC8)
 :
SNDRCVF   DEV(&DN)  RCDFMT(REC8)  WAIT(*NO)
 :
WAIT   DEV(&DN)
```

This command sends and receives user data by way of the device file named DF1. Using the record format REC8, the CL program or ILE CL procedure passes data between itself and the user who is at the device named in the variable &DN, but it does not wait for a response to come back. If the procedure sends and receives data from several devices, the same SNDRCVF command can be used. Only the device specified by &DN for the DEV parameter must be changed. A WAIT command for each device must be issued later in the procedure to ensure that all the devices respond.

Example 4: Using Open File Identifier

```
DCLF   FILE(SCREEN1)  RCDFMT(REC1 REC2)  OPNID(OUTDSP1)
DCLF   FILE(SCREEN2)  RCDFMT(REC3 REC4)  OPNID(OUTDSP2)
 :
SNDRCVF   DEV(*FILE)  RCDFMT(REC2)  OPNID(OUTDSP1)  WAIT(*YES)
```

The device file named SCREEN1 is used to send data to the display device named in the same device file and wait for input. The data is presented to the user in the format specified by record format REC2. The SNDRCVF command is associated with device file SCREEN1 because the open file identifier specified on the SNDRCVF command matches the open file identifier specified on the DCLF command for display file SCREEN1.

Top

Error messages

***ESCAPE Messages**

CPF0859
File override caused I/O buffer size to be exceeded.

CPF0861
File &1 in library &2 is not a display file.

CPF0863
Value of binary data too large for decimal CL variable.

CPF0864
End of file detected for file &1 in &2.

CPF0883
*FILE not valid in DEV parameter for file &1.

CPF0886
Record contains a data field that is not valid.

CPF0887
Data available from previous input request.

CPF4101
File &2 in library &3 not found or inline data file missing.

CPF5068
Program device &4 not found in file &2 in library &3.

CPF5070
File &2 in library &3 has no program devices acquired.

Top

Send Reply (SNDRPY)

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

Parameters
Examples
Error messages

The Send Reply (SNDRPY) command sends a reply message to the sender of an inquiry message. The message that is answered is the one having the specified message reference key that was received at the specified message queue.

A reply handling exit program can be registered in the system registration facility for exit point QIBM_QMH_REPLY_INQ. A reply handling exit program can accept, reject or replace a reply value. If the reply is rejected by an exit program, diagnostic message CPD2476 (Reply rejected by a reply handling exit program) is sent to the program running this SNDRPY command. This is followed by an escape message CPF2422 (Reply not valid). If the reply is replaced by an exit program, the send reply function sends a diagnostic message to itself. The message is CPD2479 (Reply handling exit program requested to replace a reply value). After the reply is sent, CPF2458 (Reply replaced by a reply handling exit program) will be sent as a diagnostic message and a status message to the program running this SNDRPY command. The status message can be monitored if the program needs to be aware of the condition when a reply value other than what was specified was sent.

If the specified message queue is not allocated to the job in which this command is entered, it is implicitly allocated by this command for the duration of the command.

Top

Parameters

Keyword	Description	Choices	Notes
MSGKEY	Message key	*Character value*	Required, Positional 1
MSGQ	Message queue	*Qualified object name*	Required, Positional 2
	Qualifier 1: Message queue	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
RPY	Reply	*Character value*, ***DFT**	Optional, Positional 3
RMV	Remove message	***YES**, *NO	Optional
RJTDFTRPY	Reject default reply	***NOALWRJT**, *ALWRJT	Optional
CCSID	Coded character set ID	1-65535, *HEX, ***JOB**	Optional

Top

Message key (MSGKEY)

Specifies the message reference key of the message that the reply answers.

This is a required parameter.

Top

Message queue (MSGQ)

Specifies the message queue that received the inquiry message to be answered.

This is a required parameter.

Qualifier 1: Message queue

name Specify the name of the message queue.

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 message queue. If no current library entry exists in the library list, QGPL is used.

name Specify the library where the message queue is located.

Top

Reply (RPY)

Specifies the reply that the program sends as a response to the inquiry message.

***DFT** The default reply stored in the message description of the inquiry message that was sent is sent as the reply. If no default is specified in the message description of the inquiry message, the system default reply, *N, is used.

'reply-text'
Enter the text (enclosed in apostrophes if it contains blanks or special characters) or a CL variable that contains the text that is sent as the program's reply to the inquiry message. The number of characters allowed for the reply and their format are defined by the validity specifications given in the Add Message Description (ADDMSGD) command for the specified inquiry message. However, if no validity specifications are defined for replies in the ADDMSGD command, as many as 132 characters can be used in the reply text.

Coded Character Set Identifier (CCSID) Considerations

If the inquiry message that this reply is being sent to is an immediate message, the text supplied for the RPY parameter is assumed to be in the CCSID of the job running this command unless a coded character set identifier is supplied in the CCSID parameter. If the inquiry message that this reply is sent to is a predefined message, the text supplied in the RPY parameter is assumed to be 65535 and is not converted. For more information about the message handler and its use of CCSIDs, see the i5/OS globalization topic collection in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/.

Top

Remove message (RMV)

Specifies whether the inquiry message and its reply are removed from the specified message queue.

***YES** The message and its reply are removed from the message queue when the reply is sent.

***NO** The message and its reply are held in the message queue. The inquiry message cannot be replied to more than once, but it can be received or displayed multiple times.

Top

Reject default reply (RJTDFTRPY)

Specifies whether a reply handling exit program will be allowed to reject a default reply. A default reply is requested by using blanks as the value for the reply parameter. A reply handling exit program can be registered via the system registration facility for exit point QIBM_QMH_REPLY_INQ. If a default reply is not being sent, this parameter is ignored and a reply handling exit program can reject or replace the reply value.

***NOALWRJT**
A reply handling exit program will not be allowed to reject a default reply.

***ALWRJT**
A reply handling exit program will be allowed to reject a default reply. If an exit program rejects the reply, message CPD2476 (Reply rejected by a reply handling exit program) will be sent as a diagnostic message to the program using this command. The CPD2476 will be followed by a CPF2422 (Reply not valid) escape message that the program using this command should monitor for to handle and recover from error situations.

Top

Coded character set ID (CCSID)

Specifies the coded character set identifier (CCSID) that the supplied reply text is in. If the inquiry message that this reply is sent to is a predefined message, this parameter is ignored and the text supplied for the RPY parameter is assumed to be 65535 and is not converted. For more information about the message handler and its use of CCSIDs, see the i5/OS globalization topic collection in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/.

***JOB** The reply text is assumed to be in the CCSID of the job running this command.

***HEX** The reply text is not converted. CCSID 65535 is used.

coded-character-set-identifier
Specify a valid CCSID in which you want your reply text to be considered in. Valid values range from 1 through 65535. This command will validate the CCSID. See the Globalization information in the iSeries Information Center at http://www.ibm.com/eserver/iseries/infocenter for a list of valid CCSID values.

Top

Examples

```
SNDRPY   MSGKEY(&KEY)  MSGQ(SMITH)  RPY(YES)
```

This command sends a reply of YES to the message whose reference key is specified by &KEY, which was received at message queue SMITH. Because the reply contains only one word, the reply does not have to be enclosed in apostrophes.

Top

Error messages

***ESCAPE Messages**

CPF2401
Not authorized to library &1.

CPF2403
Message queue &1 in &2 not found.

CPF2408
Not authorized to message queue &1.

CPF2410
Message key not found in message queue &1.

CPF2411
Not authorized to message file &1 in &2.

CPF2420
Reply already sent for inquiry or notify message.

CPF2422
Reply not valid.

CPF2432
Cannot send reply to message type other than *INQ or *NOTIFY.

CPF2433
Function not allowed for system log message queue &1.

CPF2460
Message queue &1 could not be extended.

CPF247E
CCSID &1 is not valid.

CPF2471
Length of field not valid.

CPF2477
Message queue &1 currently in use.

CPF2547
Damage to message file QCPFMSG.

CPF2548
Damage to message file &1 in &2.

CPF9830
Cannot assign library &1.

CPF9838
User profile storage limit exceeded.

*STATUS Messages

CPF2458
Reply replaced by reply handling exit program.

Top

Send Service Request (SNDSRVRQS)

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

Parameters
Examples
Error messages

The Send Service Request (SNDSRVRQS) command establishes a communications session and sends problem information to your service support system or tests the communications link to your service provider.

Restriction: To use this command, the user must be signed on as QSRV or QSRVBAS, or have *ALLOBJ authority.

Top

Parameters

Keyword	Description	Choices	Notes
ACTION	Action	*PREPARED, *OPENED, *TEST, *PRBID	Required, Positional 1
RMTCPNAME	Remote control point	*Communications name*, ***IBMSRV**, *SELECT	Optional
RMTNETID	Remote network identifier	*Communications name*, ***NETATR**	Optional
PRBID	Problem identifier	*Character value*	Optional

Top

Action (ACTION)

Specifies the type of request.

***PREPARED**
All records in the Problem Log with a status of PREPARED are sent to the remote service support system.

***TEST** A test is performed on the communications link to the remote service support system.

***OPENED**
All records in the Problem Log with a status of OPEN are sent to the remote service support system.

***PRBID**
A specific record from the Problem Log with a status of PREPARED or OPENED will be sent to the selected service provider.

Top

Remote control point (RMTCPNAME)

Specifies the service provider to whom the service request is sent. When *PREPARED is specified for the **Action (ACTION)** parameter, only the problem log entries that have defined destinations will be processed.

***IBMSRV**
: The service request is sent to IBM service support.

***SELECT**
: A list of service providers is shown so the user can select the destination for the service request.

remote-control-point-name
: Specify the remote control point name of the service provider to whom the request is sent.

Top

Remote network identifier (RMTNETID)

Specifies the remote name of the service provider's network.

***NETATR**
: The service provider is in the local network.

remote-network-identifier
: Specify the network name of the service provider to whom the request is being sent.

Top

Problem identifier (PRBID)

Specifies the specific problem record identifier. If the problem identifier is for a PREPARED problem, the problem must have been prepared for the selected service provider.

Top

Examples

Example 1: Sending Prepared Records

```
SNDSRVRQS   ACTION(*PREPARED)
```

This command establishes the communications link to the IBM service support system and sends all records in the problem log with the status prepared. The result of each problem log entry reported may be one of the following:

- PTFs sent to the system
- PTFs ordered from the code distribution center
- CE contacted automatically
- Service support center contacted automatically. The service support center representative will call you.
- Parts list

Example 2: Selecting a Service Provider

```
SNDSRVRQS   ACTION(*OPENED)  RMTCPNAME(*SELECT)
```

This command allows the user to select a service provider from a list. The service provider will receive all records in the problem log with an opened status.

Top

Error messages

***ESCAPE Messages**

CPF2702
Device description &1 not found.

CPF7C08
No support network connection.

CPF8C0A
Data received from &1 not recognized.

CPF8C0B
Test request routed to different system than specified.

CPF8C0C
Content of problem record &1 not valid.

CPF8C0E
Library QGPL not found.

CPF8C0F
Error indicated in reply to request.

CPF8C01
Cannot connect to IBM service system. One session allowed.

CPF8C06
No problems in problem log can be sent.

CPF8C07
A parameter is not valid.

CPF8C08
Cannot specify *SELECT for the control point name.

CPF8C09
&1 not defined as a service provider.

CPF8C16
Error occurred while processing request.

CPF8C17
Sign-on failed.

CPF8C18
No support network connection.

CPF8C19
Remote support application failed.

CPF8C2A
Cannot connect to IBM service system.

CPF8C2B
Error indicated in reply to request.

CPF8C2D
Problem &1 cannot be sent.

CPF8C24
Error occurred while processing request.

CPF8C27

Alternate load device not found.

CPF9824

Not authorized to command &1 in library &2.

CPF9846

Error while processing file &1 in library &2.

Top

Send TIE File (SNDTIEF)

Where allowed to run:

- Batch job (*BATCH)
- Batch program (*BPGM)
- Interactive program (*IPGM)
- Batch REXX procedure (*BREXX)
- Using QCMDEXEC, QCAEXEC, or QCAPCMD API (*EXEC)

Threadsafe: No

Parameters
Examples
Error messages

The Send Technical Information Exchange File (SNDTIEF) command allows you to send specified files to the remote support network.

Top

Parameters

Keyword	Description	Choices	Notes
FILE	File	*Qualified object name*	Required, Positional 1
	Qualifier 1: File	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
MBR	Member	*Name*, *FIRST	Required, Positional 2
TOUSRID	User ID	*Name*	Required, Positional 3
TONODE	Node	*Name*	Required, Positional 4
TOFILE	To file	*Name*, ***FILE**	Optional
TYPE	Type of contents of file	***OTHER**, *SYSTPLD	Optional
PTY	Priority	1, **2**	Optional

Top

File (FILE)

Specifies the physical file to be sent to the remote support network.

This is a required parameter.

The possible library values are:

***LIBL** The library list is used to locate the database file.

***CURLIB**
The current library for the job is used to locate the database file. 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 database file is located.

Top

Member (MBR)

Specifies the name of the database file member being transmitted to the remote support network.

This is a required parameter.

***FIRST**
The first member of the database file specified by the **File** prompt (FILE parameter) is transmitted.

member-name
The specified member of the database file is transmitted.

Top

User ID (TOUSRID)

Specifies the user who will receive the database file.

This is a required parameter.

Top

Node (TONODE)

Specifies the system that will receive the database file.

This is a required parameter.

Top

To file (TOFILE)

Specifies the name the database file is to have on the receiving system.

***FILE** The database file on the receiving system will have the name specified on the **File** prompt (FILE parameter).

file-name
Specify the name the database file is to have on the receiving system.

Top

Type of contents of file (TYPE)

Specifies the contents of the database file.

***OTHER**
The contents of the database file are not specified.

***SYSTPLD**
The database file contains the system configuration (topology) description.

Top

Priority (PTY)

Specifies the priority of the database file on the receiving system.

2 The data file has normal priority.

1 The file has the highest priority.

Top

Examples

```
SNDTIEF   FILE(QSYS/MYTOPO)  TOUSRID(INFSERV)  TONODE(INFTIE)
          TOFILE(ACMETOPO)
```

This command sends a file named MYTOPO from library QSYS to TIE. It is held in a mailbox for user INFSERV on system INFTIE. When it is received by the user, it is named ACMETOPO.

Top

Error messages

None

Top

Send User Message (SNDUSRMSG)

Where allowed to run: Compiled CL program or interpreted REXX (*BPGM *IPGM *BREXX *IREXX)
Threadsafe: No

Parameters
Examples
Error messages

The Send User Message (SNDUSRMSG) command is used by a program to send a message to a message queue and optionally receive a reply to that message. The message sent using this command can be either an impromptu or a predefined message and can be sent to a display station user in an interactive job or a specific message queue. For inquiry messages, a CL variable can be specified to receive the reply value, and the program using this command will wait for a response.

This command uses a combination of parameters available on the Send Program Message (SNDPGMMSG) and Receive Message (RCVMSG) commands to allow a program to send and receive messages by using a single command. Also, the SNDUSRMSG command provides validity checking and uppercase translation for replies to inquiry messages.

Restrictions:

1. The SNDUSRMSG command allows a message of up to 512 characters of first-level message text to be sent. However, if the message is sent to an external message queue (*EXT) in an interactive job, only 76 characters are shown on the Display Program Messages display. If the message is sent to a user's, work station's, or system operator's message queue, the Display Message (DSPMSG) command allows all 512 characters to be displayed.

Top

Parameters

Keyword	Description	Choices	Notes
MSG	Message text, or	*Character value*	Optional, Positional 1
MSGID	Message identifier	*Name*	Optional
MSGF	Message file	*Qualified object name*	Optional
	Qualifier 1: Message file	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
MSGDTA	Message data field values	*Character value*	Optional
VALUES	Valid reply values	Values (up to 20 repetitions): *Character value*, ***NONE**	Optional
DFT	Default reply value	*Character value*, ***MSGDFT**	Optional
MSGTYPE	Message type	***INQ**, *INFO	Optional
TOMSGQ	To message queue	Single values: *, *EXT, *SYSOPR Other values: *Qualified object name*	Optional
	Qualifier 1: To message queue	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
TOUSR	To user profile	*Name*, *SYSOPR, *REQUESTER	Optional
MSGRPY	CL var for message reply	*Character value*	Optional

Keyword	Description	Choices	Notes
TRNTBL	Translate table	Single values: *NONE Other values: *Qualified object name*	Optional
	Qualifier 1: Translate table	*Name*, **QSYSTRNTBL**	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
CCSID	Coded character set ID	1-65535, *HEX, ***JOB**	Optional

Top

Message text, or (MSG)

Specifies the message text of an immediate message that is sent by the program. A maximum of 512 characters can be specified. The text string must be enclosed in apostrophes if special characters (including blanks) are used. If this parameter is specified, values cannot be specified for the **Message identifier (MSGID)** parameter, **Message file (MSGF)** parameter, or **Message data field values (MSGDTA)** parameter.

'text' Specify the text of the immediate message to be sent.

Coded Character Set Identifier (CCSID) Considerations

The text supplied for the MSG parameter is assumed to be in the CCSID of the job running this command unless a coded character set identifier is supplied in the CCSID parameter. For more information about the message handler and its use of CCSIDs, see the i5/OS globalization topic collection in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/.

Top

Message identifier (MSGID)

Specifies the message identifier of a predefined message sent by the program to a message queue. If a value is specified for this parameter, a value must also be specified for the **Message file (MSGF)** parameter, and a value cannot be specified for the **Message text, or (MSG)** parameter.

identifier
Specify the message identifier of the message to be sent.

Top

Message file (MSGF)

Specifies the message file that contains the predefined message to be sent. This parameter is valid only if a value is specified for the **Message identifier (MSGID)** parameter.

Qualifier 1: Message file

name Specify the name of the message file which contains the predefined message to be 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 job is used to locate the message file. If no current library entry exists in the library list, QGPL is used.

name Specify the library in which the message file is located.

Top

Message data field values (MSGDTA)

Specifies the character string or the CL variable that contains a character string used as the message data in the predefined message. A character string that is specified contains one or more substitution values that are used in place of the substitution variables that were defined in the message's text when the message was defined.

***NONE**
No message data is specified for the predefined message.

message-data
Specify the character string that gives the substitution values in the specified predefined message that is sent, or specify the name of the variable that contains the character string.

Coded Character Set Identifier (CCSID) Considerations

The text supplied for the MSGDTA parameter that corresponds to the *CCHAR type field is assumed to be in the CCSID of the job running this command unless a coded character set identifier is supplied in the CCSID parameter. All other text supplied for the MSGDTA parameter is assumed to be 65535 and is not converted. For more information about the message handler and its use of CCSIDs, see the i5/OS globalization topic collection in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/. For more information about the *CCHAR type field, see the Add Message Description (ADDMSGD) command.

Top

Valid reply values (VALUES)

Specifies a list of valid replies to an inquiry message sent by this command. No more than 20 replies can be specified in the list. If the reply to the inquiry message does not match one of the specified values, an error message is sent to the reply's sender and the inquiry message is sent again.

***NONE**
No replies to inquiry messages are specified. Any reply to an inquiry message is valid.

allowable-values
Specify no more than 20 values that are compared to replies received for inquiry messages sent by this command. The maximum length of each value is 32 characters. If this value is specified, the **CL var for message reply (MSGRPY)** parameter must also be specified.

Top

Default reply value (DFT)

Specifies the value used as the reply to an inquiry message (sent by this command) if the inquiry message is sent to a message queue that is in default delivery mode, or for any other reason the default reply is sent.

***MSGDFT**
The default value defined in the message description of the message ID (specified for the MSGID parameter) is used. If no message ID is specified, the default value is *N.

'default-reply-value'
Specify the reply (enclosed in apostrophes) used as the default reply. This value can only be specified if the **CL var for message reply (MSGRPY)** parameter is specified.

Top

Message type (MSGTYPE)

Specifies the type of message to be sent.

***INQ** An inquiry message is sent and the message queue receiving the message must reply to it.

***INFO**
An informational message is sent.

Top

To message queue (TOMSGQ)

Specifies the name of the message queue to which the message is to be sent. This parameter cannot be used if a value is specified for the **To user profile (TOUSR)** parameter.

Single values

***** In an interactive job, the message is to be sent to the external message queue (*EXT). In a batch job, the message is to be sent to the system operator (message queue QSYSOPR in library QSYS).

***SYSOPR**
The message is to be sent to the system operator (message queue QSYSOPR in library QSYS).

***EXT** The message is to be sent to the job's external message queue. For batch job inquiry messages, the default reply is always received.

Qualifier 1: To message queue

name Specify the name of the message queue that is to receive the message being 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 job is used to locate the message queue. If no current library entry exists in the library list, QGPL is used.

name Specify the library where the message queue is located.

Top

To user profile (TOUSR)

Specifies that the message is to be sent to the message queue specified in the user profile for the user named on this parameter. This parameter cannot be used if a value is specified for the **To message queue (TOMSGQ)** parameter.

***SYSOPR**
The message is to be sent to the system operator user profile message queue, QSYS/QSYSOPR.

***REQUESTER**
The message is to be sent to the user profile message queue for interactive jobs or to the system operator’s message queue (QSYS/QSYSOPR) for batch jobs.

name Specify the user profile name of the user to whom the message is to be sent.

Top

CL var for message reply (MSGRPY)

Specifies the CL character variable (of up to 132 characters) that contains the reply received in response to an inquiry message. This parameter is valid only if *INQ is specified for the **Message type (MSGTYPE)** parameter.

Top

Translate table (TRNTBL)

Specifies the name of the translation table that is used if the reply value is being translated.

Qualifier 1: Translate table

QSYSTRNTBL
The translation table named QSYSTRNTBL is used to translate the reply value.

The IBM-supplied translation table QSYSTRNTBL table translates, for the English language only, all lowercase characters in the range of X’81’ to X’A9’ to uppercase characters. All other characters are not translated.

To use a different translation table, use the Create Table (CRTTBL) command and specify that particular table for this parameter.

***NONE**
The reply is not being translated.

name Specify the name of the translation table that is used to translate the message reply.

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 translation table. If no current library entry exists in the library list, QGPL is used.

name Specify the library where the translation table is located.

Top

Coded character set ID (CCSID)

Specifies the coded character set identifier (CCSID) that the message text supplied for the MSG (message) or MSGDTA (message data) parameters is in. If a message identifier is specified, the text supplied by the MSGDTA (message data) parameter that corresponds to the *CCHAR type field is assumed to be in the CCSID supplied by this parameter. The data supplied that does not correspond to the *CCHAR type field is assumed to be 65535 and is not converted. For more information about the *CCHAR type field see the Add Message Description (ADDMSGD) command.

If no message identifier is specified, the text supplied by the MSG (message) parameter is assumed to be in the CCSID supplied by this parameter. For more information about the message handler and its use of CCSIDs, see the i5/OS globalization topic collection in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/.

***JOB** The text supplied by the MSGDTA (message data) or MSG (message) parameter is assumed to be in the CCSID of the job running this command.

***HEX** The text supplied by the MSGDTA (message data) or MSG (message) parameter is not converted. CCSID 65535 is used.

coded-character-set-identifier
Specify a valid CCSID in which you want your text specified for the MSG (message) or MSGDTA (message data) parameter to be considered in. Valid values range from 1 through 65535. This command validates the CCSID. See the Globalization information in the iSeries Information Center at http://www.ibm.com/eserver/iseries/infocenter for a list of CCSID values.

Top

Examples

Example 1: Message Requiring Specific Reply

```
SNDUSRMSG   MSG('Data verified. Update master files (Y,N)?')
            TOMSGQ(*)  VALUES(Y N)  DFT(N)  MSGRPY(&REPLY)
```

This command sends an inquiry message to the display station operator (if it is used in an interactive job) or to the system operator (if it is used in a batch job). The valid replies are Y and N, and any other reply is rejected. The reply is returned in the variable &REPLY. The default translation table, QSYSTRNTBL, is used to translate the reply to uppercase characters.

Example 2: Message Requiring Any Reply

```
SNDUSRMSG   MSG('Enter any response when ready to continue.')
            TOMSGQ(WS01)
```

This command sends an inquiry message to a specific message queue. Any reply is valid. Because the purpose of this example is simply to wait, no CL variable is provided to receive the reply.

Example 3: Sending an Information Message

```
SNDUSRMSG   MSGID(USR0150)  TOUSR(FRED)  MSGF(QGPL/USRMSGF)
            MSGDTA(&ACCTNO)  MSGTYPE(*INFO)
```

This command sends a predefined message as an information message to the message queue specified in the user profile of the specified user (FRED). The message data provided is included in the message.

Top

Error messages

***ESCAPE Messages**

CPF247E

CCSID &1 is not valid.

CPF2559

Error occurred in SNDUSRMSG command.

Top

Display Mounted FS Information (STATFS)

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

Parameters
Examples
Error messages

The Display Mounted File System Information (STATFS) command displays information about a mounted file system.

This command can also be issued using the following alternative command name:
- DSPMFSINF

For more information about integrated file system commands, see the Integrated file system topic collection in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/.

Restrictions:
- The user must have execute (*X) authority to each directory in the path.

Top

Parameters

Keyword	Description	Choices	Notes
OBJ	Object	*Path name*	Required, Positional 1
OUTPUT	Output	*, *PRINT	Optional

Top

Object (OBJ)

Specifies the path name of an object that is within the mounted file system whose statistics are to be displayed. Any object in the mounted file system can be specified. For example, it can be a directory (*DIR) or a stream file (*STMF).

For more information on specifying path names, refer to ″Object naming rules″ in the CL topic collection in the Programming category in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/.

This is a required parameter.

Top

Output (OUTPUT)

Specifies whether the output from the command is displayed at the requesting work station or printed with the job's spooled output.

*
: The output is displayed for interactive jobs or printed with the job's spooled output for non-interactive jobs.

***PRINT**
The output is printed with the job's spooled output.

Top

Examples

The alternative command name for STATFS is DSPMFSINF. The following examples use the alternative command name, but STATFS can be replaced directly for DSPMFSINF in all of them.

Example 1: Displaying Statistics of a Mounted File System

```
DSPMFSINF   OBJ('/jsmith/file1')
```

This command displays the statistics for the mounted file system that contains **/jsmith/file1**.

Example 2: Displaying QSYS.LIB File System Statistics

```
DSPMFSINF   OBJ('/QSYS.LIB/MYLIB.LIB/MYFILE.FILE')
```

This command displays the statistics for the QSYS.LIB file system that contains *FILE object MYFILE in library MYLIB.

Top

Error messages

***ESCAPE Messages**

CPFA0A9
Object not found. Object is &1.

CPFA09C
Not authorized to object. Object is &1.

Top

Start Agent Services (STRAGTSRV)

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

Parameters
Examples
Error messages

The Start Agent Services (STRAGTSRV) command starts all of the agent services on this system that are defined in a preferences file. Another preferences file is also used to help define how the services are started. The preferences files are stream files named *ableplatform.preferences* and *able.preferences* that are located in the directory specified for the **Preferences file directory (PREFDIR)** parameter.

You can use the End Agent Services (ENDAGTSRV) command to end the agent services started by this command.

Restrictions:

- You must have all object (*ALLOBJ) and job control (*JOBCTL) special authorities to run this command.

Top

Parameters

Keyword	Description	Choices	Notes
PREFDIR	Preferences file directory	*Path name*, ***DFT**	Optional
HOMEDIR	Home directory	*Path name*, ***CURRENT**	Optional
CLASSPATH	Additional classpath	*Path name*, ***NONE**	Optional
SBMJOBUSER	User profile for SBMJOB	Single values: ***CURRENT** Other values (up to 10 repetitions): *Element list*	Optional
	Element 1: Pool identifier	*Character value*	
	Element 2: User profile	*Name*	

Top

Preferences file directory (PREFDIR)

Specifies the directory that contains the preferences files that define the agent services to be started and how they are started. The preferences files must be named *ableplatform.preferences* and *able.preferences*.

***DFT** Use the preferences files located in /QIBM/ProdData/OS400/able/.

path-name
Specify the directory that contains the preferences files to be used to start the agent services.

Top

Home directory (HOMEDIR)

Specifies the directory to be used as the home directory of the agent services.

***CURRENT**
Use the home directory currently in effect for the job running this command. This is the home directory defined for the current user profile of the job running the STRAGTSRV command.

path-name
Specify the path name of the directory to be used as the home directory for the Java Virtual Machines (JVMs) which are used by the agents running in these services. The services use this directory to find needed resources.

Top

Additional classpath (CLASSPATH)

Specifies the additional classpath to be appended to the classpath setting of each Java Virtual Machine (JVM) to be started to run the agent services.

***NONE**
No additional classpath will be appended.

path-name
Specify the path name of the additional classpath to be appended to the classpath setting of each JVM.

Top

User profile for SBMJOB (SBMJOBUSER)

Specifies the user profile to be used for the Java Virtual Machine (JVM) jobs that will run the agent services. You can specify that a particular JVM run under a different profile than the current user profile of the job running the STRAGTSRV command.

You can specify an alternate user profile for up to 10 JVM jobs.

Single values

***CURRENT**
All JVM jobs will run under the same user profile as the current user profile of the job running this command.

Element 1: Pool identifier

character-value
Specify an identifier that matches a JVM that is defined in the preferences file specified for the **Preferences file directory (PREFDIR)** parameter.

Element 2: User profile

name Specify the name of the user profile to be used on the Submit Job (SBMJOB) commands for the JVMs. The JVM jobs will run under the specified user profile instead of the current user profile of the job running this command.

Top

Examples

Example 1: Starting with Shipped Default Values

```
STRAGTSRV
```

This command starts all of the agent services defined in the default ableplatform.preferences file in '/qibm/prodData/OS400/able/' in a way defined in the default able.preferences file in '/qibm/prodData/OS400/able/'. The home directory of each agent service will be set to the Home Directory of the current user. All jobs submitted to run the Java Virtual Machines will run under the current profile.

Example 2: Starting with User-Specified Values

```
STRAGTSRV  PREFDIR('/qibm/userData/OS400/able/test/')
           HOMEDIR('/qibm/userData/OS400/able/test/')
           CLASSPATH('/qibm/userData/OS400/able/test/test.jar')
           SBMJOBUSER((POOL1 TESTPROF1) (POOL3 TESTPROF3))
```

This command starts all of the agent services defined in the ableplatform.preferences file in '/qibm/userData/OS400/able/test/' in a way defined in the default able.preferences file in '/qibm/userData/OS400/able/test/'. The home directory of each agent service will be set to '/qibm/userData/OS400/able/test/'. The Java Virtual Machines will have thier classpaths set to include the test.jar Java Archive file in '/qibm/userData/OS400/able/test/'. The jobs submitted to run the Java Virtual Machines for POOL1 and POOL3 will run under the users TESTPROF1 and TESTPROF3 respectivley, all other JVMs will run under the current user profile.

Top

Error messages

***ESCAPE Messages**

CPF1890
*ALLOBJ authority required for requested operation.

CPF90FF
*JOBCTL special authority required to do requested operation.

Error messages from submitted job:

This command submits a batch job which, in turn, submits additional batch jobs where the agent services will run. The following error messages could be signaled from this batch job:

CPF4B01
Agent Service not submitted. Reason code &1.

CPF4B02
The Java Virtual Machine(JVM) for the agent service was not started. Reason code &1

CPF4B03
Java Virtual Machine(JVM) exception has occurred.

Top

Start ASP Balance (STRASPBAL)

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

Parameters
Examples
Error messages

The Start ASP Balance (STRASPBAL) command allows the user to start the auxiliary storage pool (ASP) balancing function for one or more ASPs. Four types of ASP balancing can be started:

1. **Capacity balancing** - The data on all the units within the ASP will be balanced so each unit has an equal percentage of used and unused space. This is useful when new units are added to an ASP. Instead of having several units with the majority of the data and several new ones with no data, the user can spread the data evenly across all the units.
2. **Usage balancing** - The ′low′ use data on each ′low′ utilized unit in the ASP is redistributed to balance future use of the arm utilization of each unit within the specified ASP. *USAGE balancing cannot be done until the trace controlled by the Trace ASP Balance (TRCASPBAL) command has collected usage statistics. The TRCASPBAL command starts the trace function that will identify the ′high′ and ′low′ use data on each unit. After the usage balance activity has run to completion, the trace information will be cleared. Usage balancing is useful when the ASP contains large capacity disk units.
3. **Hierarchical Storage Management (HSM) balancing** - The ′high′ use and ′low′ use data on each unit in the ASP is redistributed so that the ′high′ use data resides on high performance units and the ′low′ use data resides on low performance units. The ASP being balanced in this manner must contain the correct mixture of slow and fast units to perform this type of balance. This allows the ′low′ use data on the high performance units to be moved to low performance units. Data that is ′high′ use that resides on slow units will be moved to the fast disk units. HSM balancing cannot be done until the trace controlled by the Trace ASP Balance (TRCASPBAL) command has collected usage statistics. The TRCASPBAL command starts the trace function that will identify the ′high′ and ′low′ use data on each unit. After the HSM balance activity has run to completion, the trace information will be cleared. HSM balancing is useful when the ASP contains compressed disk units.
4. **Move data from units** - This option can be used to reduce the down time associated with removing a disk unit. A unit that is scheduled for removal can be marked to end allocations by specifying UNIT(unit-number) and TYPE(*ENDALC). This will keep new allocations away from this unit. For all units marked *ENDALC, specifying TYPE(*MOVDTA) will move data from the marked units to other units in the same ASP. To resume allocations for units marked *ENDALC, specify UNIT(unit-number) and TYPE(*RSMALC). New allocations will once again be allowed to this unit. The Check ASP Balance (CHKASPBAL) command can be used to determine which units are currently marked *ENDALC.

The user may specify a time limit that the function is to run for each ASP being balanced or the balance can be set to run to completion. If the balance function needs to be ended, use the End ASP Balance (ENDASPBAL) command. A message will be sent to the system history (QHST) log when the balancing function is started for each ASP. A message will also be sent to the QHST log when the balancing function completes or is ended.

If the balance function is run for a few hours and then stopped, it will continue from where it left off when the balance function restarts. This allows the balancing to be run during off hours over a several day period.

For more information about ASP balancing, see the Hierarchical Storage Management Use, SC41-5351.

Restrictions:

- You must have all object (*ALLOBJ) special authority to run this command.

Top

Parameters

Keyword	Description	Choices	Notes
TYPE	Balance type	*CAPACITY, *USAGE, *HSM, *MOVDTA, *ENDALC, *RSMALC	Optional, Positional 2
ASP	ASP number	Single values: *ALL Other values (up to 32 repetitions): 1-32	Optional, Positional 1
ASPDEV	ASP device	Single values: *ALLAVL Other values (up to 32 repetitions): *Name*	Optional
UNIT	Storage unit	Values (up to 300 repetitions): *Integer*	Optional
TIMLMT	Time limit	1-9999, *NOMAX	Optional, Positional 3

Top

Balance type (TYPE)

Specifies the type of auxiliary storage pool (ASP) balancing operation to be done.

Note: A value must always be specified for this parameter.

***CAPACITY**
The capacity balance function will be started for the specified ASP (ASP or ASPDEV parameter).

***USAGE**
The usage balance function will be started for the specified ASP (ASP or ASPDEV parameter) .

***HSM** The hierarchical storage management balance function will be started for the specified ASP (ASP or ASPDEV parameter).

***MOVDTA**
Data will be moved off all units marked *ENDALC to other units in the same ASP. The Check ASP Balance (CHKASPBAL) command can be used to determine which units are marked *ENDALC.

***ENDALC**
New allocations will no longer go to the specific units (UNIT parameter). However, the system will use these units for new allocations to avoid ’Out of storage’ conditions.

***RSMALC**
Resume allocations to the specified units (UNIT parameter). If the unit has a much lower percentage used than other units in the ASP, run the STRASPBAL command again specifying TYPE(*CAPACITY) for the ASP that contains the unit. This will keep new allocations balanced across the units in the ASP, instead of going mostly to this unit.

Top

Auxiliary storage pool ID (ASP)

Specifies the auxiliary storage pool (ASP) for which the ASP balancing function is to be started.

Note: A value must be specified for either the **ASP number (ASP)** parameter or the **ASP device (ASPDEV)** parameter if *CAPACITY, *USAGE, or *HSM is specified for the **Balance type (TYPE)** parameter.

Single values

***ALL** ASP balancing will be started for the system ASP (ASP number 1) and all basic ASPs (ASP numbers 2-32) defined to the system.

Other values (up to 32 repetitions)

1-32 Specify the number of the ASP for which ASP balancing is to be started.

Top

ASP device (ASPDEV)

Specifies the name of the auxiliary storage pool (ASP) device for which the ASP balancing is to be started.

Note: A value must be specified for either the **ASP number (ASP)** parameter or the **ASP device (ASPDEV)** parameter if *CAPACITY, *USAGE, or *HSM is specified for the **Balance type (TYPE)** parameter.

Single values

***ALLAVL**
ASP balancing will be started for all ASP devices that currently have a status of 'Available'.

Other values (up to 32 repetitions)

name Specify the name of the independent ASP device for which ASP balancing is to be started.

Top

Storage unit (UNIT)

Specifies the unit number for which new allocations are to be ended (if *ENDALC is specified for the **Balance type (TYPE)** parameter) or for which new allocations are to resume (if *RSMALC is specified for the TYPE parameter). Specifying TYPE(*MOVDTA) will start moving data off the units marked *ENDALC.

Note: A value must be specified for this parameter if *ENDALC or *RSMALC is specified for the TYPE parameter.

integer
Specify the unit number for which new allocations are to either end (if *ENDALC is specified) or resume (if *RSMALC is specified). Up to 300 unit numbers may be specified.

Top

Time limit (TIMLMT)

Specifies the amount of time, in minutes, that the ASP balancing function will be allowed to run. When the time limit is reached, the function will end. For example, if ASP(*ALL) is specified and the machine has four ASPs configured and TIMLMT(60) is specified, four balance functions are started and each can run 60 minutes. If the balancing of any ASP has not completed after 60 minutes, it will be forced to end. This allows you to do ASP balancing incrementally. The balance function will not run across an IPL.

Note: A value must be specified for this parameter if *CAPACITY, *USAGE, *HSM, or *MOVDTA is specified for the **Balance type (TYPE)** parameter.

***NOMAX**
There is no time limit for the balance function. For a large ASP the balance function can take a long time to complete. If you start the balance function with *NOMAX and you want to force the function to end, you can use the End ASP Balance (ENDASPBAL) command.

1-9999 Specify the number of minutes that the balance function will be allowed to run.

Top

Examples

Example 1: Starting ASP Balancing for ASP 1

```
STRASPBAL   ASP(1) TIMLMT(*NOMAX) TYPE(*CAPACITY)
```

This command starts the *CAPACITY ASP balance function for ASP 1. The balance function will run until the capacity for each of the units has been balanced.

Example 2: Starting ASP Balancing with a Time Limit

```
STRASPBAL   ASP(*ALL)  TIMLMT(60)  TYPE(*USAGE)
```

This command starts the *USAGE ASP balance function for the system ASP and each configured basic ASP for which the TRCASPBAL command has been run. Each balance function will have a time limit of sixty minutes. After sixty minutes, any balance functions which have not completed will be ended.

Example 3: Starting ASP Balancing for an ASP Device

```
STRASPBAL   ASPDEV(MYASP1)  TIMLMT(*NOMAX)  TYPE(*CAPACITY)
```

This command starts the *CAPACITY ASP balance function for the ASP device named MYASP1. The balance function will run until complete.

Example 4: Preparing to Remove Units

```
STRASPBAL   UNIT(11 12 13)  TYPE(*ENDALC)
```

```
STRASPBAL   TYPE(*MOVDTA)
```

The first command marks units 11, 12 and 13 to no longer receive new allocations. The second command begins to move data off the marked units. It is recommended that the *MOVDTA ASP balancing function be done during an off-peak time.

Top

Error messages

***ESCAPE Messages**

CPF18AB
ASP balancing for ASP &1 already started.

CPF18AE
ASP &1 does not contain trace data.

CPF18B1
Trace function currently running for ASP &1.

CPF18AD
ASP &1 must contain more than a single unit.

CPF18AF
ASP &1 does not contain mixed unit types.

CPF18B3
Balance type not valid for ASP &1.

CPF1890
*ALLOBJ authority required for requested operation.

CPF9829
Auxiliary storage pool &1 not found.

Top

Start CHT Server (STRCHTSVR)

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

Parameters
Examples
Error messages

The Start Clustered Hash Table Server (STRCHTSVR) command is used to define a clustered hash table server on each cluster node specified in the NODE parameter. Successful completion of this command results in a job being started on each node defined in the cluster. Each node specified on the NODE parameter will be in the clustered hash table servers domain.

The clustered hash table server enables sharing and replicating data between cluster nodes using the Clustered Hash Table APIs. The data is stored within the clustered hash table server in non-persistent storage.

You can use the End Clustered Hash Table Server (ENDCHTSVR) command to end the clustered hash table server.

Restrictions:

- Cluster Resource Service must be active on the local node.
- All nodes specified in the NODE parameter must have Cluster Resource Services active.
- Requesting user profile must exist on all nodes specified in the NODE parameter.
- Requesting user must have change (*CHANGE) authority to the authorization list, if specified, on the current node as well as all nodes specified for the Node (NODE) parameter to perform the start.

Top

Parameters

Keyword	Description	Choices	Notes
SERVER	Server	*Communications name*	Required, Positional 1
RSPTIMO	Request response timeout	1-300, **60**	Optional
AUTL	Authorization list	*Name*, ***SERVER**	Optional
NODE	Node	Values (up to 20 repetitions): *Communications name*, ***LOCAL**	Optional

Top

Server (SERVER)

Specifies the clustered hash table server to be started.

This is a required parameter.

name Specify the name of the clustered hash table server to be started.

Top

Request response timeout (RSPTIMO)

Specifies the amount of time (in seconds) that the clustered hash table server has to complete a retrieve request. An error will be returned to the requester of the retrieve if the clustered hash table server does not get a response before the specified length of time. This parameter is ignored when the current cluster version is 3 or greater.

60 The clustered hash table server starting on the nodes specified on the NODE parameter will wait 60 seconds for a response.

1-300 Specify the number of seconds the clustered hash table server waits for a response before returning an error.

Top

Authorization list (AUTL)

Specifies an authorization list for the clustered hash table server. This defines the list of users authorized to start, end and connect to the clustered hash table server. An authorization list by the same name must exist on all nodes in the NODE parameter. The same authorization list name must be specified when starting a clustered hash table server that is already active on other nodes in the cluster.

***SERVER**

Use the same authorization list as the clustered hash table server being started if it is already active on one of the nodes in the cluster. If the server does not exist in the cluster and this value is specified then no special authority will be needed to start, end or connect to the server. This value must be specified if the current cluster version is 2 or less.

name Specify the name of the authorization list which defines the list of users authorized to the clustered hash table server. Users must have change (*CHANGE) authority to the named authorization list to start and end the clustered hash table server. Users must have use (*USE) authority to the named authorization list to connect to the clustered hash table server. Management of the authorization list is the users responsibility.

Top

Node (NODE)

Specifies the list of cluster nodes that compose the clustered hash table domain. A server job will be started on each of the cluster nodes specified. If the clustered hash table server already exists in the cluster, the cluster nodes specified will be added to the clustered hash table domain and a job will be started on the node specified. Nodes in this list must be unique.

The nodes must be active in the cluster.

***LOCAL**

A clustered hash table server job will be started on the local node only. *LOCAL can be specified only once.

name Specify the name of each cluster node that defines the clustered hash table domain. Up to 20 cluster nodes can be specified.

Top

Examples

Example 1: Starting a Local Clustered Hash Table Server

```
STRCHTSVR   SERVER(CHTSVR1)
```

This command starts clustered hash table server CHTSVR1 only on the local node. There is no authority restriction on the clustered hash table server.

Example 2: Starting a Clustered Hash Table Server on Multiple Nodes

```
STRCHTSVR   SERVER(CHTSVR2)  AUTL(AUTHLIST)  NODE(FRED BARNEY)
```

This command starts clustered hash table server CHTSVR2 on nodes FRED and BARNEY. The clustered hash table server has access restricted by authorization list AUTHLIST.

Top

Error messages

***ESCAPE Messages**

CPFBD02
Start clustered hash table server failed.

Top

Start Cleanup (STRCLNUP)

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

Parameters
Examples
Error messages

The Start Cleanup (STRCLNUP) command starts the cleanup operation, if allowed. Cleanup is allowed if *YES is specified for the **Allow cleanup (ALWCLNUP)** parameter of the Change Cleanup (CHGCLNUP) command.

A batch job is submitted to the job queue specified on the Change Cleanup (CHGCLNUP) command if cleanup is allowed. This cleanup control job submits individual batch jobs to the same job queue each day These batch jobs do the actual cleanup of the items specified on the CHGCLNUP command.

More information is in the Basic system operations topic collection in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/.

Restriction: You must have job control (*JOBCTL) special authority and have at least *USE authority to the QPGMR user profile to use this command.

Top

Parameters

Keyword	Description	Choices	Notes
OPTION	Option	***SCHED**, *IMMED	Optional, Positional 1

Top

Option (OPTION)

Specifies when the cleanup operation is started.

***SCHED**
The cleanup operation is started as scheduled for the **Time cleanup starts each day (STRTIME)** parameter of the Change Cleanup (CHGCLNUP) command or as scheduled on the Change Cleanup Options display.

***IMMED**
The cleanup operation starts immediately.

Top

Examples

Example 1: Starting Cleanup Operation as Scheduled

```
STRCLNUP
```

This command starts the cleanup operation as specified.

Example 2: Starting Cleanup Operation as Specified on CHGCLNUP Command

```
STRCLNUP   OPTION(*SCHED)
```

This command starts the cleanup operation as specified on the STRTIME parameter of the Change Cleanup (CHGCLNUP) command or on the Change Cleanup Options panel.

Example 3: Starting Cleanup Operation Immediately

```
STRCLNUP   OPTION(*IMMED)
```

This command starts the cleanup operation immediately.

Top

Error messages

***ESCAPE Messages**

CPF1E2A
Unexpected error in QSYSSCD job.

CPF1E2B
Power scheduler and cleanup options not found.

CPF1E3A
Not authorized to start cleanup.

CPF1E3C
Job queue &2/&1 not found.

CPF1E3D
Library &1 for JOBQ parameter not found.

CPF1E33
Cleanup options or power schedule in use by another user.

CPF1E34
Error occurred starting &1 job.

CPF1E36
Cleanup has not been started.

CPF1E37
Cleanup has already been started or is scheduled to run.

CPF1E38
Cleanup not allowed to run.

CPF1E39

Not authorized to job queue used for cleanup.

CPF1E99

Unexpected error occurred.

Top

Start Communications Server (STRCMNSVR)

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

Parameters
Examples
Error messages

The Start Communications Server (STRCMNSVR) command is used to start the target display station pass-through server or the display station pass-through utilities server. The target display station pass-through server processes display station pass-through, System i5 Access work station function (WSF), and other 5250 emulation programs on programmable workstations. The display station pass-through utilities server can reduce the time required to establish a session.

Restrictions You must have job control (*JOBCTL) special authority to use this command.

Top

Parameters

Keyword	Description	Choices	Notes
SERVER	Server type	***PASTHR**, *UTILSVR	Optional, Positional 1
NBRPASTHR	Number of servers	1-100, ***SYSVAL**	Optional, Positional 2

Top

Server type (SERVER)

Specifies the type of server to be started.

***PASTHR**
The target display station pass-through server is started in the QSYSWRK subsystem.

***UTILSVR**
The display station pass-through utilities server is started in the QSYSWRK subsystem.

Top

Number of servers (NBRPASTHR)

Specifies the number of target display station pass-through server jobs to be started. This parameter is only valid when *PASTHR is specified for the type of server (SERVER) parameter.

***SYSVAL**
Specifies that the system value (QPASTHRSVR) is to be used to determine the number of servers.

1-100 Specify the number of servers. This will override the current system value (QPASTHRSVR).

Top

Examples

Example 1: Starting Target Display Station Pass-through Server

```
STRCMNSVR
```

This command starts the target display station pass-through server. System value QPASTHRSVR is used to determine how many server jobs are started.

Example 2: Starting Display Station Pass-through Utilities Server

```
STRCMNSVR SERVER(*UTILSVR)
```

This command starts the display station pass-through utilities server. The time required to establish new sessions can be reduced.

Top

Error messages

***ESCAPE Messages**

CPF8946
Unexpected error ending target display station pass-through servers.

Top

Start Communications Trace (STRCMNTRC)

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

Parameters
Examples
Error messages

The Start Communications Trace (STRCMNTRC) command initiates a communications trace for a specified line, a network interface or a network server description.

A communications trace continues until:

- The End Communications Trace (ENDCMNTRC) command is run.
- The Communications Trace function of the Start System Service Tools (STRSST) command is used to end the trace.
- A physical line problem causes the trace to end.
- TRCFULL(*STOPTRC) is specified and the buffer becomes full.
- Automatically by the watch for trace event functionality.

Restrictions:

- The user must have *USE authority to the line, network interface, or network server to be traced.
- To use this command, you must have service (*SERVICE) special authority, or be authorized to the Service trace function of i5/OS through System i 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
  - QSRV
- When the **Watched job (WCHJOB)** parameter is specified, the issuer of the command must be running under a user profile which is the same as the job user identity of the job being watched, or the issuer of the command must be running under a user profile which has job control (*JOBCTL) special authority. Job control (*JOBCTL) special authority is also required if a generic user name is specified for the WCHJOB parameter.
- If you specify a generic user name in the WCHJOB parameter, you must have all object (*ALLOBJ) special authority, or be authorized to the Watch any job function of i5/OS through System i Navigator's Application Administration support. The Change Function Usage (CHGFCNUSG) command, with a function ID of QIBM_WATCH_ANY_JOB, can also be used to change the list of users that are allowed to start and end watch operations.
- You must have operational (*OBJOPR) and execute (*EXECUTE) authorities to the user exit program if specified in **Trace program (TRCPGM)** parameter, and execute (*EXECUTE) authority to the library where the program is located.
- You must have use (*USE) authority to the message queues specified in **Watched message queue (WCHMSGQ)** parameter, and use (*USE) authority to the library where the message queue is located.

Top

Parameters

Keyword	Description	Choices	Notes
CFGOBJ	Configuration object	*Name*	Required, Positional 1
CFGTYPE	Type	*LIN, *NWI, *NWS	Required, Positional 2
MAXSTG	Buffer size	*Integer*, *MIN, *MAX, **128K**, 256K, 2M, 4M, 6M, 8M, 16M, 32M, 64M, 128M, 256M, 512M, 1G	Optional
DTADIR	Data direction	*SND, *RCV, ***BOTH**	Optional
TRCFULL	Trace full	***WRAP**, *STOPTRC	Optional
USRDTA	Number of user bytes to trace	Single values: ***CALC**, *MAX Other values: *Element list*	Optional
	Element 1: Beginning bytes	*Decimal number*	
	Element 2: Ending bytes	*Decimal number*, *CALC	
CMNTRCOPTS	Communications trace options	***ALLDTA**, *RMTCTL, *RMTMAC, *RMTSAP, *LCLSAP, *IPPCLNUM, *RMTIPADR	Optional
DDITRCOPTS	DDI trace options	*ALLDTA, *RMTCTL, *RMTMAC, *RMTSAP, *LCLSAP, *IPPCLNUM, *RMTIPADR	Optional
RMTCTL	Remote controller	*Name*	Optional
RMTMAC	Remote MAC address	*Hexadecimal value*	Optional
RMTSAP	Remote SAP	*Hexadecimal value*	Optional
LCLSAP	Local SAP	*Hexadecimal value*	Optional
IPPCLNUM	IP protocol number	0-255, *ICMP, *IGMP, *TCP, *EGP, *IGP, *UDP	Optional
RMTIPADR	Remote IP address	*Character value*	Optional
LMITRCOPTS	LMI trace options	***ALLDTA**, *NOLMI, *LMIONLY	Optional
NWSTRCOPTS	NWS trace options	***NETBIOS**, *INTERNAL, *TCPIP	Optional
WCHMSG	Watch for message	Single values: ***NONE** Other values (up to 5 repetitions): *Element list*	Optional
	Element 1: Message identifier	*Name*	
	Element 2: Comparison data	*Character value*, ***NONE**	
	Element 3: Compare against	***MSGDTA**, *FROMPGM, *TOPGM	
WCHMSGQ	Watched message queue	Values (up to 3 repetitions): *Element list*	Optional
	Element 1: Message queue	Single values: ***SYSOPR**, *JOBLOG, *HSTLOG Other values: *Qualified object name*	
	Qualifier 1: Message queue	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**	
WCHJOB	Watched job	Single values: * Other values (up to 5 repetitions): *Element list*	Optional
	Element 1: Job name	*Qualified job name*	
	Qualifier 1: Job name	*Generic name, name*	
	Qualifier 2: User	*Generic name, name*	
	Qualifier 3: Number	000001-999999, ***ALL**	

Keyword	Description	Choices	Notes
WCHLICLOG	Watch for LIC log entry	Single values: ***NONE** Other values (up to 5 repetitions): *Element list*	Optional
	Element 1: Major code	*Character value*, *ALL	
	Element 2: Minor code	*Character value*, *ALL	
	Element 3: Comparison data	*Character value*, ***NONE**	
	Element 4: Compare against	***ALL**, *TDENBR, *TASKNAME, *SVRTYPE, *JOBNAME, *JOBUSR, *JOBNBR, *THDID, *EXCPID, *MODNAME, *MODRUNAME, *MODEPNAME, *MODOFFSET, *MODTSP	
WCHPAL	Watch for PAL entry	Single values: ***NONE** Other values (up to 5 repetitions): *Element list*	Optional
	Element 1: System reference code	*Character value*, *ALL	
	Element 2: Comparison data	*Character value*, ***NONE**	
	Element 3: Compare against	***RSCNAME**, *RSCTYPE, *RSCMODEL	
WCHTIMO	Length of time to watch	1-43200, ***NOMAX**	Optional
TRCPGM	Trace program	Single values: ***NONE** Other values: *Qualified object name*	Optional
	Qualifier 1: Trace program	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**	
TRCPGMITV	Time interval	1-9999, ***NONE**	Optional
RUNPTY	Run priority	1-99, **25**	Optional
TEXT	Trace description	*Character value*, ***BLANK**	Optional

Top

Configuration object (CFGOBJ)

Specifies the configuration object to be traced. The object is either a line description, a network interface description, or a network server description.

name Specify the name of the configuration object to be traced.

Top

Type (CFGTYPE)

Specifies the type of configuration description to trace.

***LIN** The configuration object is a line description.

***NWI** The configuration object is a network interface description.

***NWS** The configuration object is a network server description.

Top

Buffer size (MAXSTG)

Specifies the trace buffer size.

128K A trace buffer of 128 kilobytes is used.

***MIN** The minimum trace buffer size is used.

***MAX** The maximum trace buffer size is used.

buffer-size
Specify the trace buffer size. Valid buffer sizes may be specified either as number of kilobytes, or as one of the following special values which has a one-letter suffix of ’K’ for kilobytes, ’M’ for megabytes, or ’G’ for gigabytes: 128K, 256K, 2M, 4M, 6M, 8M, 16M, 32M, 64M, 128M, 256M, 512M, 1G. The minimum trace buffer size is 128 kilobytes.

Top

Data direction (DTADIR)

Specifies the communication data to trace.

Note: For network server description traces, this parameter is ignored and *BOTH is used.

***BOTH**
Data sent and received by the system is traced.

***SND** Data sent by the system is traced.

***RCV** Data received by the system is traced.

Top

Trace full (TRCFULL)

Specifies the action the system takes when the trace buffer is full of data.

***WRAP**
The trace continues and overwrites the data in the buffer.

***STOPTRC**
The trace stops.

Top

Number of user bytes to trace (USRDTA)

Specifies the amount of beginning and ending user data to trace.

Note: For network server description traces and binary synchronous lines, this parameter is ignored and *CALC is used.

Single values

***CALC**
The system determines the number of beginning and ending bytes to be traced. For LAN lines, this is the first 100 bytes. For other line types, the whole frame is traced.

***MAX** Trace as much of frames as possible. For non-LAN, *MAX will be the equivalent of *CALC.

Element 1: Beginning bytes

decimal-number
Specify the number of bytes of beginning user data to be traced.

Element 2: Ending bytes

***CALC**
The system determines the number of ending bytes to be traced.

decimal-number
Specify the number of bytes of ending user data to be traced.

Top

Communications trace options (CMNTRCOPTS)

Specifies the type of data to be traced.

***ALLDTA**
All data is traced. No filtering is specified.

***RMTCTL**
The data traveling to and from a remote controller is traced.

***RMTMAC**
The data traveling to and from a remote medium access control (MAC) address is traced.

***RMTSAP**
The data traveling to and from a remote service access point (SAP) is traced.

***LCLSAP**
The data traveling to and from a local service access point (SAP) is traced.

***IPPCLNUM**
The data within an Internet Protocol (IP) number is traced.

***RMTIPADR**
The data traveling to and from a remote IP address is traced.

Top

DDI trace options (DDITRCOPTS)

The DDITRCOPTS parameter is supported for upward compatibility of CL programs which contain the STRCMNTRC command. The CMNTRCOPTS parameter provides all of the same function as DDITRCOPTS and should be used instead of DDITRCOPTS.

***ALLDTA**
All data is traced. No filtering is specified.

***RMTCTL**
The data traveling to and from a remote controller is traced.

***RMTMAC**
The data traveling to and from a remote medium access control (MAC) address is traced.

***RMTSAP**
The data traveling to and from a remote service access point (SAP) is traced.

***LCLSAP**
The data traveling to and from a local service access point (SAP) is traced.

***IPPCLNUM**
The data within an Internet Protocol (IP) number is traced.

***RMTIPADR**
The data traveling to and from a remote IP address is traced.

Top

Remote controller (RMTCTL)

Specifies the remote controller receiving and sending the data to be traced.

name Specify the name of the remote controller.

Top

Remote MAC address (RMTMAC)

Specifies the remote medium access control address receiving and sending the data to be traced.

hexadecimal-value
Specify the remote medium access control address.

Top

Remote SAP (RMTSAP)

Specifies the remote service access point receiving and sending the data to be traced.

hexadecimal-value
Specify the remote service access point.

Top

Local SAP (LCLSAP)

Specifies the local service access point receiving and sending the data to be traced.

hexadecimal-value
Specify the local service access point.

Top

IP protocol number (IPPCLNUM)

Specifies the Internet Protocol (IP) number to be traced.

***ICMP**
The Internet control message group is traced.

***IGMP**
The Internet group management group is traced.

***TCP** The transmission control group is traced.

***EGP** The exterior gateway protocol group is traced.

***IGP** A private interior gateway group is traced.

***UDP** The user datagram group is traced.

0-255 Specify the Internet Protocol (IP) number to trace.

Top

Remote IP address (RMTIPADR)

Specifies the remote Internet Protocol (IP) address to be traced.

character-value
Specify the remote IP address to be traced.

Top

LMI trace options (LMITRCOPTS)

Specifies the type of data to be placed in the trace buffer.

***ALLDTA**
All data, including the local management interface (LMI), is placed in the trace buffer.

***NOLMI**
All data, except LMI data, is placed in the trace buffer.

***LMIONLY**
Only LMI data is placed in the trace buffer.

Top

NWS trace options (NWSTRCOPTS)

Specifies the type of data to be placed in the trace buffer.

***NETBIOS**
All NetBIOS data is placed in the trace buffer.

***INTERNAL**
The communications processor operating system data is placed in the trace buffer.

***TCPIP**
All TCP/IP data for network server description applications is placed in the trace buffer.

Top

Watch for message (WCHMSG)

Specifies up to five message identifiers which are to be watched for. If a value other than *NONE is specified, you must specify where to watch for the message on the WCHMSGQ parameter. When the watched for message is added to the specified message queue or log, the trace exit program is called; if no trace exit program is defined, the trace stops.

Single values

***NONE**
No messages will be watched for.

Element 1: Message identifier

name Specify the 7-character message identifier to be watched for.

Element 2: Comparison data

Specify comparison data to be used if a message matching the specified message ID is added to the specified message queue or log. If the message data, the ″From program″ or the ″To program″ includes the specified text, the watched for condition is true. If the message data, the ″From program″ or the ″To program″ does not contain the specified text, the trace function continues.

***NONE**
No comparison data is specified. If a message matching the specified message ID is added to the specified message queue or log, the watched for condition is true.

character-value
Specify the text string used to compare against the message data, the ″From program″ or the ″To program″ of the watched for message. This text is case sensitive and can be quoted in order to specify imbedded or trailing blanks.

Element 3: Compare against

Specify which part of the message the comparison data specified for element 2 is to be compared against.

***MSGDATA**
The comparison data will be compared against the message replacement data.

***FROMPGM**
The comparison data will be compared against the name of the program sending the message, or the name of the ILE program that contains the procedure sending the message.

***TOPGM**
The comparison data will be compared against the name of the program the message was sent to, or the name of the ILE program that contains the procedure the message was sent to.

Top

Watched message queue (WCHMSGQ)

Specifies where to watch for the message identifiers specified on the WCHMSG parameter. You can specify to watch the message being added to the system operator message queue, the history log, other message queues, and job logs. Up to three message queues or special values can be specified.

Element 1: Message queue

Single values

***SYSOPR**
Watch messages added to the system operator message queue (QSYSOPR message queue in library QSYS).

***JOBLOG**
Watch messages added to the job logs of the jobs specified for the **Watched job (WCHJOB)** parameter.

***HSTLOG**
Watch messages added to the history log (QHST message queue in library QSYS).

Qualifier 1: Message queue

name Specify the name of the message queue to watch.

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 where the message queue is located.

Top

Watched job (WCHJOB)

Specifies the job whose job log is watched for the messages specified on the WCHMSG parameter. The specified job will only be watched if *JOBLOG is specified on the WCHMSGQ parameter. Up to five job names may be specified.

Single values

_*_ Only the job log of the job that issued this trace command is watched.

Element 1: Job name

Qualifier 1: Job name

generic-name
Specify the generic name of the job to be watched. 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 job name specifies all jobs with job names that begin with the generic prefix.

name Specify the name of the job to be watched.

Qualifier 2: User

generic-name
Specify the generic name of the user name of the job to be watched. 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 user name specifies all jobs with the specified job name and with user names that begin with the generic prefix.

name Specify the user name of the job to be watched.

Qualifier 3: Number

*ALL All jobs with the specified job name and user name are watched.

000001-999999
Specify the job number to further qualify the job name and user name. You cannot specify a job number if a generic job name or a generic user name qualifier is specified.

Top

Watch for LIC log entry (WCHLICLOG)

Specifies up to five licensed internal code (LIC) log entry identifiers which are to be watched for. Each LIC log entry contains a major and a minor code. The watched for condition will be met if a LIC log entry is added that matches the specified major and minor codes and any comparison data specified. When the watched for log entry is added to the LIC log, the trace exit program is called, even when the comparison data specified does not match; if no trace exit program is defined, the trace stops.

Single values

***NONE**
No LIC log entries will be watched for.

Element 1: Major code

***ALL** Any LIC log entry major code will be considered to be a match. If *ALL is specified for the major code, you cannot specify *ALL for the LIC log entry minor code.

character-value
Specify the LIC log major code to be watched for. You can specify either a hexadecimal digit or a question mark for each character in the four-digit code. A question mark is a wildcard character that will match any digit in that position. Up to three wildcard characters can be specified.

Element 2: Minor code

***ALL** Any LIC log entry minor code will be considered to be a match. If *ALL is specified for the minor code, you cannot specify *ALL for the LIC log entry major code.

character-value
Specify the LIC log minor code to be watched for. You can specify either a hexadecimal digit or a question mark for each character in the four-digit code. A question mark is a wildcard character that will match any digit in that position. Up to three wildcard characters can be specified.

Element 3: Comparison data

Specify comparison data to be used if a log entry matching the specified major and minor codes is added to the licensed internal code (LIC) log. If this text is found in the LIC log entry data fields of the watched for log entry, the watched for condition is true. If this text is not found in the LIC log entry data fields of the watched for log entry and no exit program is specified on the TRCPGM parameter, the trace function continues. If the log entry matches the specified major and minor codes and an exit program is specified on the TRCPGM parameter, but the entry data does not contain the specified text, the exit program is called to determine if the trace should continue or stop.

***NONE**
No comparison data is specified. If a LIC log entry matching the specified major and minor codes is added to the LIC log, the watched for condition is true.

character-value
Specify the text string used to compare against the entry data of the watched for log entry. If this text is found in the LIC log entry data field specified for element 4, the watch condition is considered to be true. This text is case sensitive. If *ALL is specified in the LIC log compare against field, the LIC log fields which will be compared are TDE number, task name, server type, job name, user ID, job number, thread ID, exception ID, LIC module compile timestamp, LIC module offset, LIC module RU name, LIC module name, LIC module entry point name. The comparison data cannot be used to match across two fields, and can match an entire field or a substring of any field.

When watching for an exception ID, all four hexadecimal digits of the exception ID must be specified. Also, the prefix MCH may be specified if you want to compare only against the exception ID field and avoid possible substring matches with the other fields.

Element 4: Compare against

Specify which part of the LIC log the comparison data specified for element 3 is to be compared against.

***ALL** The LIC log comparison data will be compared against all the fields described below.

***TDENBR**
The LIC log comparison data will be compared against the number of the task dispatching element (TDE) which requested the LIC log entry.

***TASKNAME**
The LIC log comparison data will be compared against the name of the task which requested the LIC log entry. Task name is blank (hex 40s) if the LIC log entry is not requested by a task.

***SVRTYPE**
The LIC log comparison data will be compared against the type of server that requested the LIC log entry. Server type is blank (hex 40s) if the LIC log entry is not requested by a server.

***JOBNAME**
The LIC log comparison data will be compared against the name of the job which requested the LIC log entry. LIC job name is blank (hex 40s) if the LIC log entry is not requested by a job.

***JOBUSR**
The LIC log comparison data will be compared against the user name of the job which requested the LIC log entry. LIC user name is blank (hex 40s) if the LIC log entry is not requested by a job.

***JOBNBR**
The LIC log comparison data will be compared against the job number (000001-999999) to further qualify the job name and user name of the job which requested the LIC log entry. LIC job number is blank (hex 40s) if the LIC log entry is not requested by a job.

***THDID**
The LIC log comparison data will be compared against the thread which requested the LIC log entry. Thread identifier is binary zeros if the LIC log entry is not requested by a thread.

***EXCPID**
The LIC log comparison data will be compared against the exception that caused the LIC log entry to be requested. This is a 2-byte hexadecimal field formed by concatenating to the high-order 1-byte exception group number a low-order 1-byte exception subtype number. Exception identifier is binary zeros if the LIC log entry is not requested as a result of an exception.

***MODNAME**
The LIC log comparison data will be compared against the LIC module name which requested the LIC log entry. If the module name is greater than 64 characters, the LIC module name is truncated to 64 characters.

***MODRUNAME**
The LIC log comparison data will be compared against the LIC module replacement unit name. LIC module RU name is always in upper case EBCDIC.

***MODEPNAME**
The LIC log comparison data will be compared against the name of the entry point which requested the LIC log entry. If the entry point name is greater than 128 characters, the LIC module entry point name is truncated to 128 characters.

***MODOFFSET**
The LIC log comparison data will be compared against the byte offset into the LIC module text which requested the LIC log entry.

***MODTSP**
The LIC log comparison data will be compared against the timestamp of when the LIC module was compiled. The format for this field is the system time-stamp format.

Top

Watch for PAL entry (WCHPAL)

Specifies up to five Product Activity Log (PAL) entries which are to be watched for. When the watched for PAL occurs, the trace exit program is called; if no trace exit program is defined, the trace stops.

Single values

***NONE**
No PAL entries will be watched for.

Other values (up to 5 repetitions)

Element 1: System reference code

***ALL** Any system reference code will be considered to be a match.

character-value
Specify the system reference code (SRC) to be watched for. You can specify either a hexadecimal digit or a question mark for each character in the eight-digit code. A question mark is a wildcard character that will match any digit in that position. Up to seven wildcard characters can be specified. You can also specify a generic SRC that 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 SRC specifies all PAL entries with system reference codes that begin with the generic prefix.

Element 2: Comparison data

Specify comparison data to be used if a PAL entry matching the specified system reference code occurs. If the field specified in element 3 matches the specified text, the watched for condition is true. If the field specified in element 3 does not match the specified text, the watch function just continues.

***NONE**
No comparison data is specified. If a PAL entry matching the specified system reference code occurs, the watched for condition is true.

character-value
Specify the text string used to compare against the field specified in element 3 of the watched for PAL entry. This text is case sensitive.

You can specify question mark (?) and asterisk (*) wildcard characters in the text string. A question mark is a single-character wildcard and will match any character in the same position. For example, ’??123’ will match any value that is five characters long and ends with ’123’. Multiple question mark wildcard characters can be specified for the comparison data value.

An asterisk is a multiple-character wildcard character. You can specify a single asterisk wildcard character at the end of the comparison data value. For example, ’ABC*’ will match any value that begins with the letters ’ABC’.

Element 3: Compare against

Specify which part of the PAL entry the comparison data specified for element 2 is to be compared against.

***RSCNAME**
The comparison data will be compared against the name of the physical device that has the entry in the log. A resource name is assigned at first by the system, but may have been changed to a new value by a user.

***RSCTYPE**
The comparison data will be compared against the number or word used to identify a product.

***RSCMODEL**
The comparison data will be compared against the numbers or letters used to identify the feature level of a product with a given type.

Top

Length of time to watch (WCHTIMO)

Specifies the time limit, in minutes, for watching for a message or a licensed internal code (LIC) log entry or a Product Activity Log (PAL) entry. When the specified amount of time has elapsed, the trace exit program is called (if one was specified on the TRCPGM parameter), the trace is ended, and message CPI3999 is sent to the history log.

***NOMAX**
There is no time limit for watching for a particular message or LIC log entry or PAL entry.

1-43200
Specify the number of minutes that the trace will remain active while none of the watched for conditions have been met.

Top

Trace program (TRCPGM)

Specifies the program to be called for user-defined trace commands and procedures.

The trace program will be called:

- Before the application trace starts.
- After a match of a message identifier specified for the WCHMSG parameter, or a match of a Licensed Internal Code (LIC) log entry specified for the WCHLICLOG parameter, or a match of a Product Activity Log (PAL) entry specified for the WCHPAL parameter occurs.
- When the time interval specified on the TRCPGMITV parameter is reached.
- When the length of time to watch specified on WCHTIMO parameter is reached.

There are three input parameters and one output parameter associated with the trace program. The four parameters are required:

```
1   Trace option setting   Input    Char(10)
2   Reserved               Input    Char(10)
3   Error detected         Output   Char(10)
4   Comparison data        Input    Char(*)
```

Allowed values for the ″Trace option setting″ parameter are:

***ON** The watch for trace facility is starting when the collection of trace information is started.

***MSGID**
A match on a message id specified on WCHMSG parameter occurred.

***LICLOG**
A match on a LIC log specified on the WCHLICLOG parameter occurred.

***CMPDATA**
The major and minor code of a LIC log matched, but the comparison data did not.

***INTVAL**
The time interval specified on TRCPGMITV parameter is elapsed.

***WCHTIMO**
The length of time to watch specified on WCHTIMO parameter is elapsed.

***PAL** A match on a PAL and any associated comparison data specified on the WCHPAL parameter occurred.

The ″Reserved″ parameter must be set to blanks.

Allowed values for the ″Error detected″ parameter are:

***CONTINUE**
The trace and the watch for trace event facility will continue running.

***STOP**
The trace and the watch for trace event facility will be ended.

***ERROR**
Error detected by customer trace program.

Allowed values for the ″Comparison data″ parameter when *MSGID is specified for the ″Trace option setting″ parameter will be the following structure:

```
OFFSET     TYPE         FIELD
Dec Hex
 0   0     BINARY(4)    Length of trace information
 4   4     CHAR(7)      Message ID
11   B     CHAR(9)      Reserved
20  14     BINARY(4)    Offset to comparison data
24  18     BINARY(4)    Length of comparison data
 *   *     CHAR(*)      Message comparison data
```

Allowed values for the ″Comparison data″ parameter when *LICLOG or *CMPDATA is specified for the ″Trace option setting″ parameter will be the following structure:

```
OFFSET     TYPE         FIELD
Dec Hex
 0   0     BINARY(4)    Length of trace information
 4   4     CHAR(4)      LIC Log major code
 8   8     CHAR(4)      LIC Log minor code
12   C     CHAR(8)      LIC Log identifier
20  14     BINARY(4)    Offset to comparison data
24  18     BINARY(4)    Length of comparison data
 *   *     CHAR(*)      LIC log comparison data
```

Allowed values for the ″Comparison data″ parameter when *ON, *INTVAL or *WCHTIMO is specified for the ″Trace option setting″ parameter will be the following structure:

```
OFFSET     TYPE         FIELD
Dec Hex
 0   0     BINARY(4)    Length of trace information (always 4).
```

Allowed values for the ″Comparison data″ parameter when *PAL is specified for the ″Trace option setting″ parameter will be the following structure:

```
OFFSET     TYPE         FIELD
Dec  Hex
  0    0   BINARY(4)    Length of watch information
  4    4   CHAR(8)      System reference code
 12    C   CHAR(10)     Device name
 22   16   CHAR(4)      Device type
 26   1A   CHAR(4)      Model
 30   1E   CHAR(15)     Serial number
 45   2D   CHAR(10)     Resource name
 55   37   CHAR(8)      Log identifier
 63   3F   CHAR(8)      PAL timestamp
 71   47   CHAR(4)      Reference code
 75   4B   CHAR(8)      Secondary code
 83   53   CHAR(8)      Table identifier
 91   5B   CHAR(1)      Reserved
 92   5C   BINARY(4)    Sequence
 96   60   BINARY(4)    Offset to comparison data
 100  64   BINARY(4)    Length of comparison data
 104  68   CHAR(10)     PAL compare against
 *   *     CHAR(*)      PAL comparison data
```

For more information on the trace exit program interface, refer to the APIs topic collection in the Programming category in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/

Single values

***NONE**
No trace exit program is defined. If a watched for message or licensed internal code (LIC) log entry or Product Activity Log (PAL) entry is added, or if the specified watch time limit is exceeded, the trace function ends.

Qualifier 1: Trace program

name Specify the name of the trace exit program.

Qualifier 2: Library

***LIBL** All libraries in the job's library list are searched until the first match is found.

name Specify the name of the library where the user exit program is located.

Top

Time interval (TRCPGMITV)

Specifies how often the trace exit program will be called.

***NONE**
No time interval is specified. The trace exit program will not be called because a time interval has elapsed.

1-9999 Specify the interval of time, in seconds, of how often the trace exit program will be called. This must be less than the amount of time specified for the **Length of time to watch (WCHTIMO)** parameter.

Top

Run priority (RUNPTY)

Specifies the priority of the job where the watch session work will be run.

25
A job priority of 25 will be used.

1-99
Specify the run priority of the job. For more information on job run priority, refer to the Work management topic collection in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/

Top

Trace description (TEXT)

Specifies the text that briefly describes the object.

***BLANK**
Text is not specified.

character-value
Specify up to 20 characters of text.

Top

Examples

Example 1: Start a Communications Trace for a Line Description

```
STRCMNTRC   CFGOBJ(*QESLINE)  CFGTYPE(*LIN)
```

This command starts a communications trace of line description QESLINE.

Example 2: Start a Trace and Watch for a Message to End the Trace

```
STRCMNTRC   CFGOBJ(LINE001)  CFGTYPE(*LIN)  WCHMSG((MCH2804))
            WCHMSGQ((*SYSOPR) (*JOBLOG))
            WCHJOB((*ALL/MYUSER/MYJOBNAME))
            TRCPGM(MYLIB/TRCEXTPGM)
```

This command starts a communications trace of line description LINE001. The trace will be ended when MCH2804 message is found on the System Operator message queue or within the *ALL/MYUSER/MYJOBNAME job log. Also, MYLIB/TRCEXTPGM is specified as a trace exit program.

Example 3: Start a Trace and Watch for a LIC Log Entry to End the Trace

```
STRCMNTRC   CFGOBJ(LINE001)  CFGTYPE(*LIN)
            WCHLICLOG(('99??' 9932 MYJOBNAME))
            WCHTIMO(*NOMAX)
```

This command starts a communications trace of line description LINE001. The trace will be ended when a Licensed Internal Code (LIC) log entry that has a major code starting with 99 and a minor code of 9932 is generated on the system. Also, the LIC log information should contain the text ″MYJOBNAME″. *NOMAX on WCHTIMO parameter indicates that the trace will be active until the event occurs or ENDCMNTRC command is issued manually.

Top

Error messages

***ESCAPE Messages**

CPF2601
Line description &1 not found.

CPF2634
Not authorized to object &1.

CPF39AA
Trace &1 type &2 already exists

CPF39AB
Beginning or ending bytes exceeds maximum value

CPF39AC
Total of beginning and ending bytes exceeds maximum value

CPF39AD
&1 type &2 cannot be traced

CPF39A6
Storage could not be allocated

CPF39A7
Trace storage not available in communications processor

CPF39A8
Not authorized to communications trace service tool

CPF39A9
Error occurred during communications trace function

CPF39BD
Network interface description &1 not found

CPF39BF
Remote IP address not valid.

CPF39B6
Communications trace function cannot be performed

CPF39C0
Controller description &1 not found.

CPF39C1
Controller description &1 not valid.

CPF39C2
Number of user bytes to trace must be *CALC.

CPF39F1
Trace buffer size too large.

CPF39F2
Cannot allocate library &1

CPF98A2
Not authorized to &1 command or API.

Top

Start Commitment Control (STRCMTCTL)

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

Parameters
Examples
Error messages

The Start Commitment Control (STRCMTCTL) command is used to establish either a job level or activation group level commitment definition. The job's current name space determines which independent auxiliary storage pool (IASP) the commitment definition is created in. Files can be opened under commitment control for this commitment definition either when the files reside in that same ASP or when the files reside in *SYSBAS and the commitment definition has no resources registered.

This command also specifies the level of record locking that occurs for the commitment definition to be started. Also, a notify object can be specified.

Before a commitment definition is established, the user must ensure that all database files that are to be opened under commitment control for a single commitment transaction are journaled. If only the after images are being journaled, the system implicitly begins journaling both the before and the after images for the duration of the changes being made to files opened under this commitment definition.

A default journal can be specified. Entries that describe all journals and systems involved in a commitment control operation can be placed in this journal.

More information on the use of journal management is in the Journal management topic collection in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/. More information on journaling related to commitment control is in the Commitment control topic collection in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/.

Restrictions:

1. The user must have object operational and add authority to the object named on the NFYOBJ parameter, if an object is specified.
2. The user must have object operational and add authority to the object named on the DFTJRN parameter, if an object is specified.

Top

Parameters

Keyword	Description	Choices	Notes
LCKLVL	Lock level	*CHG, *CS, *ALL	Required, Positional 1
NFYOBJ	Notify object	Single values: ***NONE** Other values: *Element list*	Optional, Positional 2
	Element 1: Object	*Qualified object name*	
	Qualifier 1: Object	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
	Element 2: Object type	Single values: *MSGQ, *DTAARA Other values: *Element list*	
	Element 1: (*MSGQ *DTAARA or *FILE)	*FILE	
	Element 2: Member, if *FILE	*Name*, ***FIRST**	

Keyword	Description	Choices	Notes
CMTSCOPE	Commitment definition scope	***ACTGRP**, *JOB	Optional
TEXT	Text 'description'	*Character value*, ***DFTTEXT**	Optional
DFTJRN	Journal	Single values: ***NONE** Other values: *Qualified object name*	Optional
	Qualifier 1: Journal	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
OMTJRNE	Journal entries to be omitted	***NONE**, *LUWID	Optional

Top

Lock level (LCKLVL)

Specifies the default level of record locking that occurs for the commitment definition to be started.

This is a required parameter.

***CHG** Every record read for update (for a file opened under commitment control) is locked. If a record is changed, added, or deleted, that record remains locked until the transaction is committed or rolled back. Records that are accessed for update operations but are released without being changed are unlocked.

***CS** Every record accessed for files opened under commitment control is locked. A record that is read, but not changed or deleted, is unlocked when a different record is read. Records that are changed, added, or deleted are locked until the transaction is committed or rolled back.

***ALL** Every record accessed for files opened under commitment control is locked until the transaction is committed or rolled back.

Top

Notify object (NFYOBJ)

Specifies the name and type of the object where notification is sent regarding the status of a transaction for a commitment definition. The commitment identifier of the last successful commit operation is sent to the notify object only for the following conditions:

- For a job level commitment definition, if any of the following are true:
  - A system failure occurs
  - The job ends with uncommitted changes
  - The job ends with a nonzero completion code
- For an activation group level commitment definition, if any of the following are true:
  - A system failure occurs
  - The job ends with uncommitted changes
  - The job ends with a nonzero completion code
  - The activation group ends abnormally
  - The activation group ends with uncommitted changes and the uncommitted changes are rolled back

For a system failure, the commitment identifier is placed in the notify object after the next successful initial program load (IPL). For a job that ends with uncommitted changes or with a nonzero completion code, the commitment identifier is placed in the notify object during end job processing. For an activation group that ends with uncommitted changes or ends abnormally, the notification text is placed in the notify object during activation group end processing.

A commitment identifier (specified for the **Commit identification (CMTID)** parameter on the Commit (COMMIT) command) can be specified on each commit operation performed for a commitment definition. If more than one job is concurrently using commitment control or there is more than one commitment definition being used concurrently within a single job, then each commitment definition for each job should use a unique notify object or the specified commit identifier should contain unique text such that the text identifies a single commitment definition for a single job. If *NONE is specified for the CMTID parameter of the Commit (COMMIT) command, this entry is ignored.

***NONE**
No notification is sent after an abnormal system or process end.

object-name
Specify the name (library-name/object-name) of the object to receive notification of the last transaction that is successfully committed. You must have correct authority for the object specified.

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

library-name
Specify the library where the object is located.

The possible object type values are:

***MSGQ**
The text identifying the last commitment boundary is placed on the specified message queue.

***DTAARA**
The text identifying the last commitment boundary is placed in the specified data area. The data area specified must be of type character, and unique to this job. The text is padded or truncated to fit the data area.

***FILE** The text identifying the last commitment boundary is added to the specified physical file.

The possible physical file member values are:

***FIRST**
The first member of the physical file receives the notification.

member-name
Specify the name of the member of the physical file that receives the notification.

Top

Commitment definition scope (CMTSCOPE)

Specifies the scope for the commitment definition to be started.

***ACTGRP**
An activation-group-level commitment definition is started for the activation group associated with the program issuing the command.

***JOB** The job-level commitment definition is started for the job.

Top

Text 'description' (TEXT)

Specifies text that briefly describes the commitment definition to be started. More information on this parameter is in the CL Reference book, Appendix A.

***DFTTEXT**
The system is to provide a default text description for the commitment definition.

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

Top

Journal (DFTJRN)

Specifies the default journal. The default journal contains entries identifying each of the resources involved in a unit of work. Entries can also be placed when each unit of work starts or ends due to a commit or rollback operation, depending on the OMTJRNE parameter value.

More information on the default journal is in the Commitment control topic collection in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/.

The default journal can be used when adding a resource through the Add Committable Resource (QTNADDCR) Application Program Interface (API). If the special value *DFTJRN is specified for the journal name when calling the API, the name specified on this DFTJRN parameter is used.

***NONE**
No default journal is specified.

The name of the default journal can be qualified by one of the following 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.

journal-name
Specify the name of the default journal.

Top

Journal entries to be omitted (OMTJRNE)

Specifies the journal entries to omit from the default journal. If *NONE is specified on the DFTJRN parameter, this parameter is ignored.

***NONE**
No journal entries are omitted.

***LUWID**
The journal entry that contains the Logical Unit of Work Identifier (LUWID) and all the resources

involved in the logical unit of work are omitted if the logical unit of work is committed or rolled back successfully. If an error occurs while committing or rolling back the logical unit of work, the entry will always be sent regardless of this value.

Top

Examples

Example 1: Defining Activation Group Level Commitment Control

```
STRCMTCTL   LCKLVL(*CHG)  CMTSCOPE(*ACTGRP)  TEXT('Blue Commit Group')
```

This command described by the user as the Blue Commit Group starts the activation group level commitment for the activation group associated with the program issuing the command.

Only records that are updated, inserted, or deleted are locked until the transaction is ended by a commit or rollback operation. No identification for the commitment boundary is sent after the initial program load (IPL) following an abnormal system end, after an abnormal end to an activation group for the job, or when the job or activation group ends either with uncommitted changes or with a nonzero completion code.

Example 2: Defining Job Level Commitment Control

```
STRCMTCTL   LCKLVL(*ALL)  NFYOBJ(RCVLIB/MYFILE *FILE IDSAVE)
            CMTSCOPE(*JOB)  DFTJRN(MGWLIB/MYJRN)
```

This command starts the job level commitment definition. All records accessed in files opened under commitment control are locked until the commitment transaction is ended by a commit or rollback operation. If a commitment transaction ends in a manner that a notify object is to be updated with the commitment identifier of the last successful commit operation, the notify object to be updated is member IDSAVE of file MYFILE in the library RCVLIB. When a commit or rollback is done, an entry that lists information about all the resources involved in the logical unit of work is put into journal MYJRN in library MGWLIB.

Top

Error messages

***ESCAPE Messages**

CPF8351
Commitment control already active.

CPF8352
Attribute in notify object &1 type *&4 not valid.

CPF8360
Not enough storage for commitment control operation.

CPF8366
Commitment definition &2 not created. Reason code &1.

CPF9801
Object &2 in library &3 not found.

CPF9802
Not authorized to object &2 in &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.

CPF9815
Member &5 file &2 in library &3 not found.

CPF9820
Not authorized to use library &1.

CPF9830
Cannot assign library &1.

Top

Start Copy Screen (STRCPYSCN)

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

Parameters
Examples
Error messages

The Start Copy Screen (STRCPYSCN) command allows you to copy the screens of another display station on your display station to observe what is happening and diagnose problems.

If the STRCPYSCN command is used to copy displays from a source device that has the wide-display feature to an output device with a regular-width display, the command is accepted, but wide-display images are not shown and an informational message is sent to the target work station indicating that the display was not shown.

If the STRCPYSCN command is used to copy displays from a source device that supports graphic DBCS characters, the command is accepted and character information is shown, but graphic DBCS characters appear as single byte. No message is sent.

If the output device is not the requesting device, then the output device cannot be signed on. If the output device is signed on, a message is sent to the requester indicating that the device is not available for copying. If the source device is signed off after display copy has begun, the function automatically ends.

Note: The copy display function can be ended by the target device.

Restrictions:

- Permission must be given from the user of the source work station.
- When a request is made to begin display image copying, a break message is sent to the user of the source work station to inform the user that the displays are going to be copied. The user must reply to this message before any displays are copied.
- *REQUESTER is not valid for the SRCDEV or OUTDEV parameters when the command is submitted to batch.

Top

Parameters

Keyword	Description	Choices	Notes
SRCDEV	Source device	*Name*, *REQUESTER	Required, Positional 1
OUTDEV	Output device	*Name*, *REQUESTER, *NONE	Required, Positional 2
JOBQ	Job queue	*Qualified object name*	Optional
	Qualifier 1: Job queue	*Name*, **QCTL**	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
OUTFILE	File to receive output	Single values: ***NONE** Other values: *Qualified object name*	Optional
	Qualifier 1: File to receive output	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	

Keyword	Description	Choices	Notes
OUTMBR	Output member options	*Element list*	Optional
	Element 1: Member to receive output	*Name*, ***FIRST**	
	Element 2: Replace or add records	***REPLACE**, *ADD	

Top

Source device (SRCDEV)

Specifies the display station that is used as the source for the screen images to be copied.

This is a required parameter.

***REQUESTER**
Specifies that the screens are to be copied from the display station that issued this command.

name Specify the name of the display station (other than the one that issued this command) whose screens are to be copied.

Top

Output device (OUTDEV)

Specifies the output device for the copying process.

This is a required parameter.

***REQUESTER**
Specifies that screens are to be copied to the work station from which this command is issued. *REQUESTER cannot be specified here if it is also specified for the **Source device (SRCDEV)** parameter.

***NONE**
Specifies that the copied screens do not go to a display station. If *NONE is specified here, then a value must be specified for the **File to receive output (OUTFILE)** parameter.

name Specify the name of the display station (other than the one that issued this command) that will display the copied screens.

Top

Job queue (JOBQ)

Specifies the job queue used to submit the job which shows the screens from the source device on the target device when the requesting device is not the target device. When *REQUESTER is specified on the **Output device (OUTDEV)** parameter, this parameter is ignored, since it defaults to the values for the target display station and then a submit job is not necessary.

Qualifier 1: Job queue

QCTL Job queue QCTL is to be used.

name Specify the name of the job queue where the job which will process the copied screens is to be submitted.

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 queue. If no current library entry exists in the library list, QGPL is used.

name Specify the name of the library where the job queue is located.

Top

File to receive output (OUTFILE)

Specifies the database file to which the output of the command is directed. If the file does not exist, this command creates a database file in the specified library. If the file is created, the public authority for the file is the same as the create authority specified for the library in which the file is created. Use the Display Library Description (DSPLIBD) command to show the library's create authority.

Qualifier 1: File to receive output

name Specify the name of the database file to which the command output is directed.

Qualifier 2: Library

***LIBL** The library list is used to locate the file. If the file is not found, one is created in the current library. If no current library exists, the file will be created in the QGPL library.

***CURLIB**
The current library for the thread is used to locate the 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.

Note: If a new file is created, system file QASCCPY in system library QSYS with a format name of QSCCPY1 is used as a model.

Top

Output member options (OUTMBR)

Specifies the name of the database file member that receives the output of the command.

Element 1: Member to receive output

***FIRST**
The first member in the file receives the output. If OUTMBR(*FIRST) is specified and the member does not exist, the system creates a member with the name of the file specified for the **File to receive output (OUTFILE)** parameter. If the member already exists, you have the option to add new records to the end of the existing member or clear the member and then add the new records.

name Specify the name of the file member that receives the output. If it does not exist, the system creates it.

Element 2: Replace or add records

***REPLACE**
The system clears the existing member and adds the new records.

***ADD** The system adds the new records to the end of the existing records.

Top

Examples

```
STRCPYSCN   SRCDEV(WS2)  OUTDEV(*REQUESTER)
```

This command sends an inquiry message to the user of work station, WS2. The message indicates that the display station displays are about to be copied to another display station. If the user of that display does not wish this to happen, then a cancel (C) reply prevents the operation from beginning. To allow the operation to begin, the user responds with a go (G) reply to the message.

Top

Error messages

***ESCAPE Messages**

CPF2207
Not authorized to use object &1 in library &3 type *&2.

CPF7AF4
Library QTEMP is not valid for OUTFILE keyword.

CPF7AF5
From device cannot be used with to device.

CPF7AF6
Device &1 not available.

CPF7AF7
Device name &1 not correct.

CPF9845
Error occurred while opening file &1.

CPF9860
Error occurred during output file processing.

***NOTIFY Messages**

CPI7AF9
Screen image not displayed.

Top

Start Debug (STRDBG)

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

Parameters
Examples
Error messages

The Start Debug (STRDBG) command puts a job into debug mode and, optionally, adds as many as 20 programs and 20 service programs and 20 class files to debug mode. It also specifies certain attributes of the debugging session. For example, it can specify whether database files in production libraries can be updated while in debug mode.

The graphical system debugger will be launched instead of the traditional system debugger if the STRDBG command is issued by a user registered for graphical debugging.

Debug can operate in three environments at the same time. They are OPM (Original Program Model), ILE (Integrated Language Environment), and JAVA. Some parameters for this command are applicable for all three environments. Some parameters for this command are applicable only for OPM, ILE, or JAVA. There will be a statement in each parameter description stating the environment in which the parameter is applicable.

The Change Debug (CHGDBG) command can be used later in the job to change the attributes of the debug mode. Also, OPM programs can be added to or removed from the debugging session if they are specified in the Add Program (ADDPGM) or Remove Program (RMVPGM) commands. OPM programs added with the ADDPGM command will be in the OPM debug environment.

When one job is servicing another job, and STRDBG is entered, all debug commands are valid for the job being serviced. If that job is held on a job queue, no further debug commands may be entered until that job is allowed to run. When the job starts, an initial breakpoint screen is displayed. From this screen, additional debug commands may be entered. To service another job, see the STRSRVJOB (Start Service Job) command. More information about debugging one job from another job is in the CL topic collection in the Programming category in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/.

Restrictions:

- You cannot use this command in debug mode. To end debug mode, refer to the End Debug (ENDDBG) command.
- You cannot use this command if you are servicing another job, and that job is held, suspended, or ending.
- This command is shipped with public *EXCLUDE authority and the QPGMR, QSRV, and QSRVBAS user profiles have private authorities to use this command.
- You must have either *CHANGE authority to the program, or *USE authority to the program and *SERVICE special authority.

Top

Parameters

Keyword	Description	Choices	Notes
PGM	Program	Single values: ***NONE** Other values (up to 20 repetitions): *Qualified object name*	Optional, Positional 1
	Qualifier 1: Program	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
DFTPGM	Default program	*Name*, ***PGM**, *NONE	Optional
MAXTRC	Maximum trace statements	*Integer*, **200**	Optional
TRCFULL	Trace full	***STOPTRC**, *WRAP	Optional
UPDPROD	Update production files	***NO**, *YES	Optional
OPMSRC	OPM source level debug	***NO**, *YES	Optional
SRVPGM	Service program	Single values: ***NONE** Other values (up to 20 repetitions): *Qualified object name*	Optional
	Qualifier 1: Service program	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
CLASS	Class file	Values (up to 20 repetitions): *Path name*, ***NONE**	Optional
DSPMODSRC	Display module source	***PGMDEP**, *NO, *YES	Optional
SRCDBGPGM	Source debug program	Single values: ***SYSDFT** Other values: *Qualified object name*	Optional
	Qualifier 1: Source debug program	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
UNMONPGM	Unmonitored message program	Single values: ***NONE** Other values: *Qualified object name*	Optional
	Qualifier 1: Unmonitored message program	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	

Top

Program (PGM)

Specifies up to 20 programs to debug in the job. Before a program can be debugged, its name must be specified on this parameter, in the Add Program (ADDPGM) command, or the Display Module Source (DSPMODSRC) command.

This parameter is applicable in both OPM and ILE environments.

Single values

***NONE**
No program names are specified at the start of the debugging session. The Add Program (ADDPGM) command or the Display Module Source (DSPMODSRC) command can be used to add programs later.

Other values (up to 20 repetitions)

Qualifier 1: Program

name Specify the name of the program to debug. You cannot debug two programs that have the same name at the same time in the OPM debug environment. In the ILE debug environment, two programs with the same name can be debugged if they are in different libraries.

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 library where the program is located.

Top

Default program (DFTPGM)

Specifies the original program model (OPM) program to use as the default program during debug mode. The program specified here is used as the default program for any of the other debug commands for which the user specified a value of *DFTPGM for the **Program (PGM)** parameter. That is, if a default program was previously specified, this parameter can change it.

This parameter is applicable only in the OPM environment.

***PGM** The program named on the PGM parameter of this command is to be the default program for the job's debugging session. If there is more than one program name specified on the PGM parameter, the first OPM program named in the list that is added to the OPM debug environment is the default program. If *NONE is specified on the PGM parameter, or is the default, *NONE is also assumed when this value (*PGM) is specified.

***NONE**
No program is specified as the default program; if a program was specified as a default program, it is no longer the default program. If the job has no default program, *DFTPGM cannot be specified on the PGM parameter of any other debug commands.

name Specify the name of the program to use as the default program during debug mode. The same name (in qualified form) must also be specified for the PGM parameter of this command. You cannot specify a bound program on this parameter.

Note: If OPMSRC(*YES) is specified, the OPM program may be added to either the OPM debug environment or the ILE debug environment. If added to the ILE debug environment, then it cannot be a default program for the OPM debug environment.

Top

Maximum trace statements (MAXTRC)

Specifies the maximum number of trace statements that the system puts into the job's trace file before either stopping tracing or wrapping around (overlaying) on the trace file. When the trace file contains the maximum specified, the system performs the actions specified by the value supplied for the **Trace full (TRCFULL)** parameter.

Note: Instruction stepping can be performed on a program being debugged in an interactive environment by setting the maximum number of trace statements to **1** and the value for the TRCFULL parameter to *STOPTRC.

This parameter is applicable only in the OPM environment.

200 Two hundred trace statements can be put into the file before tracing is stopped or wrapping occurs.

integer
Specify the maximum number of trace statements that can be in the trace file.

Top

Trace full (TRCFULL)

Specifies what happens when the job's trace file is full (that is, it contains the maximum number of trace statements specified by the value supplied for the **Maximum trace statements (MAXTRC)** parameter.

This parameter is applicable only in the OPM environment.

***STOPTRC**
In a batch environment, tracing stops but the program continues processing. In an interactive environment, control is given to the user when a breakpoint occurs. If the user continues processing, a breakpoint occurs before processing each subsequent statement within the range of statements being traced, and the trace file is extended to contain the new entry.

***WRAP**
The trace file is overlaid with new trace statements as they occur, wrapping from the beginning of the file. The program will complete processing with no message to indicate that wrapping has occurred. The trace file never has more than the maximum specified statements, and they are the most recently recorded statements.

Top

Update production files (UPDPROD)

Specifies whether or not database files in a production library can be opened for updating records, or for adding new records, while the job is in debug mode. If not, the files must be copied into a test library before trying to run a program that uses the files.

This parameter is applicable in both OPM and ILE environments.

***NO** Database files in production libraries cannot be updated while the job is in debug mode. Database files can be opened for reading only. This protects database files from unwanted updates while a program is being debugged. The exception to this is starting debug mode after a production library is already opened. If this value is specified, some commands, which may include function keys, menu options, or pulldowns, may not work properly.

***YES** Database files in production libraries can be updated while the job is in debug mode.

Top

OPM source level debug (OPMSRC)

Specifies that original program model (OPM) programs are debugged using the system source debug support (same as ILE source debug).

This parameter is valid for OPM CL, OPM RPG and OPM COBOL programs that were created with OPTION(*SRCDBG) with CRTCLPGM, CRTRPGPGM and CRTCBLPGM commands. Additionally, this parameter is valid for OPM CL, OPM RPG and OPM COBOL programs that were created with OPTION(*LSTDBG) with the CRTCLPGM, CRTRPGPGM, CRTCBLPGM, CRTSQLRPG, CRTSQLCBL, and CRTRPTRPG commands. If the OPM program is not CL, RPG or COBOL and was not compiled with a valid debug option, then this parameter is ignored.

This parameter is applicable in both OPM and ILE environment.

ILE functions will now operate on OPM programs. For example, source level debug.

***NO** OPM debug functions are used for OPM programs.

***YES** ILE debug functions are used for OPM programs.

Top

Service program (SRVPGM)

Specifies up to 20 service programs to debug in the job. Before a service program can be debugged, its name must be specified on this parameter or the Display Module Source (DSPMODSRC) command.

This parameter is applicable only in the ILE environment.

Single values

***NONE**
No service program names are specified at the start of the debugging session. The Display Module Source (DSPMODSRC) command can be used to add service programs later.

Other values (up to 20 repetitions)

Qualifier 1: Service program

name Specify the name of the service program to debug.

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 service program. If no library is specified as the current library for the job, QGPL is used.

name Specify the library where the service program is located.

Top

Class file (CLASS)

Specifies up to 20 class files to debug in the job. Before a class file can be debugged, its name must be specified on this parameter or the Display Module Source (DSPMODSRC) command.

This parameter is applicable only in the ILE environment.

***NONE**
No class files are specified at the start of the debugging session. The Display Module Source (DSPMODSRC) command can be used to add class files later.

path-name
Specify the path names of the class file to debug.

Top

Display module source (DSPMODSRC)

Specifies whether the first display of the source debug program is shown when this command is processed and no errors occur.

This parameter is applicable only in the ILE environment.

***PGMDEP**
The showing of the source debug program display is dependent on the programs specified on the **Program (PGM)** parameter. If any of the programs are ILE programs, the display is shown. If any of the programs are OPM programs with source debug data, and OPMSRC(*YES) is specified, the display is shown. If all of the programs are OPM programs and OPMSRC(*NO) is specified, the display is not shown.

***NO** The first display of the source debug program is not shown.

***YES** The first display of the source debug program is shown.

Top

Source debug program (SRCDBGPGM)

Specifies the source debug program to be used. See the APIs topic collection in the Programming category in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/ for an explanation of this parameter.

This parameter is applicable only in the ILE environment.

Single values

***SYSDFT**
The system source debug program is used.

Qualifier 1: Source debug program

name Specify the name of the program to be used to debug programs.

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.

Top

Unmonitored message program (UNMONPGM)

Specifies the qualified name of the user-supplied program called when a message that is not monitored occurs in the job being debugged. When the program specified is called, it is passed parameters that identify the program name, the recursion level, the high-level language statement identifier, the machine instruction number at which the breakpoint occurred, the message that was not monitored, the message data, the length of the message data, and the message reference key.

This parameter is applicable only in the OPM environment.

The passed parameters have the following formats:

1. Program name (10 bytes). Specifies the name of the program in which the breakpoint was reached.
2. Recursion level (5 bytes). Specifies the recursion level number of the program in which the breakpoint was reached. This value is a 1- to 5-digit number padded on the right with blanks.
3. Statement Identifier (10 bytes). Specifies the high-level language program statement identifier that was reached. This is the statement identifier specified on the Add Breakpoint (ADDBKP) command. If a machine instruction number is used to specify the breakpoint, this parameter contains a slash (/) followed by a 4-digit hexadecimal machine instruction number.
4. Instruction number (5 bytes). Specifies the machine instruction number that corresponds to the high-level language statement at which the breakpoint was reached. No slash appears in front of the machine instruction number. The value consists of 1 to 4 hexadecimal characters representing the MI instruction number, followed by one or more blanks. If a machine instruction number is passed on the third parameter, the numbers in the third and fourth parameters are the same.
5. Message ID (7 bytes). Specifies the ID of the message that was not monitored.
6. Message data (256 bytes). Specifies the first 256 bytes of message data sent with the message not monitored.
7. Message data length (5 bytes). Specifies the length of the message data sent with the message not monitored.
8. Message MRK (4 bytes). Specifies the message reference key (MRK) of the message not monitored.

All the parameter values are left-adjusted and padded on the right with blanks. When control returns to the program with the message that was not monitored, processing continues.

Single values

***NONE**
No program is called when a message that is not monitored occurs.

Qualifier 1: Unmonitored message program

name Specify the name of the user-supplied program called when a message that is not monitored occurs in the job being debugged. After the program runs, control is returned to the interrupted 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 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.

Top

Examples

Example 1: Start Debug for Program

```
STRDBG   PGM(TESTLIB/PAYROLL)  UPDPROD(*NO)
```

This command starts debug mode to debug the program PAYROLL, which is in the test library TESTLIB. If tracing is used, up to 200 trace statements can be stored in the trace before tracing stops. If program

PAYROLL is a bound program, the Display Module Source display will be shown, giving the source for the module that contains the program entry point. Any database files updated by the PAYROLL program must be in a test library.

Example 2: Start Debug for Class File

```
STRDBG   CLASS('financial.payapplet')
```

This command starts debug mode to debug the class **payapplet** in the **financial** package.

Top

Error messages

***ESCAPE Messages**

CPF1999
Errors occurred on command.

Top

Start Debug Server (STRDBGSVR)

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

Parameters
Examples
Error messages

The Start Debug Server (STRDBGSVR) command starts the debug server router function. Only one debug server router can be active at a time. Once started, the debug server router remains active until it is ended using the ENDDBGSVR command.

There are no parameters for this command.

Top

Parameters

None

Top

Examples

```
STRDBGSVR
```

This command starts the debug server router function.

Top

Error messages

None

Top

Start Database Monitor (STRDBMON)

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

Parameters
Examples
Error messages

The Start Database Monitor (STRDBMON) command starts the collection of database performance statistics for a specified job, for all jobs on the system or for a selected set of jobs. The statistics are placed in a user-specified database file and member. If the file or member do not exist, one is created based on the QAQQDBMN file in library QSYS. If the file or member do exist, the record format of the specified file is verified to insure it is the same.

For each monitor started using the STRDBMON command, the system generates a monitor ID that can be used to uniquely identify each individual monitor. The monitor ID can be used on the ENDDBMON command to uniquely identify which monitor is to be ended. The monitor ID is returned in the informational message CPI436A which is generated for each occurrence of the STRDBMON command. The monitor ID can also be found in column QQC101 of the QQQ3018 database monitor record.

Restrictions:

1. There are two types of monitors. A private monitor is a monitor over one, specific job (or the current job). A public monitor is a monitor which collects data across multiple jobs. Only one (1) monitor can be started on a specific job at a time (i.e. only one private monitor can be active over any specific job). For example, STRDBMON JOB(*) followed by another STRDBMON JOB(*) within the same job is not allowed. There can be a maximum of ten (10) public monitors active at any one time. For example, STRDBMON JOB(*ALL) followed by another STRDBMON JOB(*ALL) is allowed providing the maximum number of public monitors does not exceed 10. You may have 10 public monitors and 1 private monitor active at the same time for any specific job.
2. If multiple monitors specify the same output file, only one copy of the database statistic records will be written to the specified output file for each job. For example, STRDBMON OUTFILE(LIB/TABLE1) JOB(*) and STRDBMON OUTFILE(LIB/TABLE1) JOB(*ALL) both use the same output file. For the current job, you will not get two copies of the database statistic records, one copy for the private monitor and one copy for the public monitor. You will get only one copy of the database statistic records.
3. QTEMP cannot be specified as the library on the OUTFILE parameter unless JOB(*) was also specified.
4. This command is conditionally threadsafe. For multithreaded jobs, this command is not threadsafe and may fail when the OUTFILE parameter is a distributed file or is a Distributed Data Management (DDM) file.
5. Any public monitor requires the file specified for the OUTFILE parameter to be in a library that resides in the system ASP.

Top

Parameters

Keyword	Description	Choices	Notes
OUTFILE	File to receive output	*Qualified object name*	Required, Positional 1
	Qualifier 1: File to receive output	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	

Keyword	Description	Choices	Notes
OUTMBR	Output member options	*Element list*	Optional
	Element 1: Member to receive output	*Name*, ***FIRST**	
	Element 2: Replace or add records	***REPLACE**, *ADD	
JOB	Job name	Single values: ***** Other values: *Qualified job name*	Optional
	Qualifier 1: Job name	*Generic name, name*, *ALL	
	Qualifier 2: User	*Generic name, name*, *ALL	
	Qualifier 3: Number	000000-999999, *ALL	
TYPE	Type of records	***BASIC**, *DETAIL, *SUMMARY	Optional
FRCRCD	Force record write	0-32767, ***CALC**	Optional
RUNTHLD	Run time threshold	0-2147483647, ***NONE**	Optional
STGTHLD	Storage threshold	0-2147483647, ***NONE**	Optional
INCSYSSQL	Include system SQL	***NO**, *YES, *INI	Optional
FTRFILE	Filter by database file	Single values: ***NONE** Other values (up to 10 repetitions): *Qualified object name*	Optional
	Qualifier 1: Filter by database file	*Generic name, name*, *ALL	
	Qualifier 2: Library	*Generic name, name*	
FTRUSER	Filter by user profile	*Generic name, name*, ***NONE**, *CURRENT	Optional
FTRINTNETA	Filter by internet address	*Character value*, ***NONE**, *LOCAL	Optional
FTRLCLPORT	Filter by local port number	0-65535, ***NONE**	Optional
FTRQRYGOVR	Filter by query governor	***NONE**, *ALL, *COND	Optional
COMMENT	Comment	*Character value*, ***BLANK**	Optional

Top

File to receive output (OUTFILE)

Specifies the file to which the performance statistics are to be written. If the file does not exist, it is created based on model file QAQQDBMN in library QSYS.

This is a required parameter.

Qualifier 1: File to receive output

name Specify the name of the file.

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.

Top

Output member options (OUTMBR)

Specifies the name of the database file member that receives the output of the command.

Element 1: Member to receive output

***FIRST**
The first member in the file receives the output. If OUTMBR(*FIRST) is specified and the member does not exist, the system creates a member with the name of the file specified for the **File to receive output (OUTFILE)** parameter. If the member already exists, you have the option to add new records to the end of the existing member or clear the member and then add the new records.

name Specify the name of the file member that receives the output. If it does not exist, the system creates it.

Element 2: Replace or add records

***REPLACE**
The system clears the existing member and adds the new records.

***ADD** The system adds the new records to the end of the existing records.

Top

Job name (JOB)

Specifies the job(s) for which the database monitor is to be started.

Single values

***** The database monitor for the job running the STRDBMON command is to be started.

***ALL** All jobs on the system are monitored, including jobs waiting on job queues.

Qualifier 1: Job name

name Specify the name of the job whose database monitor is to be started. If no job user name or job number qualifiers are specified, all of the jobs currently in the system are searched for the specified simple job name. If duplicates of the specified job name are found, you need to specify a job user name or job number that uniquely identifies the job to be changed.

generic-name
Specify the generic name of the jobs whose database monitor are to be started. All jobs matching the specified generic name, including jobs waiting on job queues, will be monitored.

Qualifier 2: User

name Specify the name of the user of the job whose database monitor is to be started.

generic-name
Specify the generic name of the user whose jobs are to be monitored. All jobs matching the specified generic name, including jobs waiting on job queues, will be monitored.

Qualifier 3: Number

000000-999999
Specify the number of the job whose database monitor is to be started.

Top

Type of records (TYPE)

Specifies the type of database records to place in the outfile.

***BASIC**
Only the basic database monitor records are collected.

***DETAIL**
Both basic and detail database monitor records are collected. The detail database monitor record (QQQ3019) contains a count of the number of synchronous and asynchronous reads and writes to the database, as well as other database counts.

***SUMMARY**
Only the basic database monitor records are collected.

Top

Force record write (FRCRCD)

Specifies the number of records to be held in the buffer before forcing the records to be written to the file when running with a private monitor.

***CALC**
The system will calculate the number of records to be held in the buffer.

number-of-records
Specify the number of records to be held. Valid values range from 0 through 32767.

Top

Run time threshold (RUNTHLD)

Specifies a filtering threshold based on the estimated run time of the SQL statement as calculated by the query optimizer. Monitor records will be created only for those SQL statements whose estimated run time meets or exceeds the specified run time threshold. If the estimated run time of the SQL statement is less than the specified threshold then no monitor records will be created for that SQL statement.

***NONE**
A run time threshold is not specified. All monitor records will be created.

runtime-threshold
Specify the run time threshold, in seconds. Monitor records will be created for all SQL statements whose estimated run time meets or exceeds this value.

Top

Storage threshold (STGTHLD)

Specifies a filtering threshold based on the estimated temporary storage usage of the SQL statement. Monitor records will be created only for those SQL statements whose estimated temporary storage meets or exceeds the specified storage threshold. If the estimated temporary storage of the SQL statement is less than the specified threshold then no monitor records will be created for that SQL statement.

***NONE**
A storage threshold is not specified. All monitor records will be created.

storage-threshold
Specify the storage threshold, in megabytes. Monitor records will be created for all SQL statements whose estimated temporary storage meets or exceeds this value.

Top

Include system SQL (INCSYSSQL)

Specifies whether or not monitor records will be created for system-generated SQL statements. Monitor records will always be created for user-specified SQL statements. This option determines if monitor records will also be created for SQL statements generated internally by the system.

***NO** No monitor records will be created for system-generated SQL statements. Monitor records will only be created for user-specified SQL statements.

***YES** Monitor records will be created for both user-specified and system-generated SQL statements.

***INI** Monitor records will be generated based on the value of the SQL_DBMON_OUTPUT option in the current INI file. A value of *USER or *DEFAULT creates monitor records for just user-specified SQL statements. A value of *SYSTEM creates monitor records for just system-generated SQL statements. A value of *ALL creates monitor records for both user-specified and system-generated SQL statements. If no INI file exists, then a default value of *NO will be used for the INCSYSSQL option.

Top

Filter by database file (FTRFILE)

Specifies a filter based on the name of the file and library used within the SQL statement. Monitor records will be created only for those SQL statements that use the qualified file. The specified file name can be either the 10-byte short name or the 256-byte long name.

Single values

***NONE**
No filtering by file is specified.

Qualifier 1: Data base file

***ALL** Monitor records will be created for any SQL statement that uses any file in the specified library. If none of the files used in the SQL statement come from the specified library, no monitor records will be created for the SQL statement.

name Monitor records will be created only for those SQL statements that use the specified file. Monitor records will not be created for any SQL statements that do not use the specified file.

generic-name
Monitor records will be created only for those SQL statements that use a file that matches the generic prefix. If none of the files used in the SQL statement match the specified prefix, no monitor records will be created for the SQL statement.

Qualifier 2: Library

name Monitor records will be created only for those SQL statements that use a file from the specified library. Monitor records will not be created if none of the files used in the SQL statement come from the specified library.

generic-name
Monitor records will be created only for those SQL statements that use a file from a library that matches the generic prefix. If none of the files used in the SQL statement come from the generic library, no monitor records will be created for the SQL statement.

Top

Filter by user profile (FTRUSER)

Specifies a filter based on a user profile name. Monitor records will be created only for those SQL statements that are executed by the specified user. Monitor records will not be created for SQL statements executed by a different user.

***NONE**
Filtering by user is not specified.

***CURRENT**
Monitor records will be created only for those SQL statements that are executed by the user who is invoking the STRDBMON command. Monitor records will not be created for SQL statements executed by a different user.

user-name
Monitor records will be created only for those SQL statements that are executed by the specified user. Monitor records will not be created for SQL statements executed by a different user.

generic-user-name
Monitor records will be created only for those SQL statements that are executed by a user whose name starts with the specified prefix. Monitor records will not be created for SQL statements executed by a different user.

Top

Filter by internet address (FTRINTNETA)

Specifies filtering based on the client's TCP/IP address or client's TCP/IP host name. Monitor records will be created only for jobs running on behalf of the specified TCP/IP address or host name.

***NONE**
Internet address filtering is not specified.

***LOCAL**
Monitor records will be created for those jobs that have no TCP/IP client association.

character-value
Monitor records will be created for TCP/IP database server jobs. Jobs named QRWTSRVR and QZDASOINIT are examples of these server jobs.

1. IP version 4 address in dotted decimal form. Specify an internet protocol version 4 address in the form nnn.nnn.nnn.nnn where each nnn is a number in the range 0 through 255.
2. IP version 6 address in colon hexadecimal form. Specify an internet protocol version 6 address in the form xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx:xxxx where each xxxx is a hex number in the range 0 through FFFF. IP version 6 includes the IPv4-mapped IPv6 address form (for example, ::FFFF:1.2.3.4). For IP version 6, the compressed form of the address is allowed.
3. IP host domain name. Specify an internet host domain name of up to 254 characters in length.

Top

Filter by local port number (FTRLCLPORT)

Specifies filtering based on the local TCP/IP port number. Monitor records will be created for TCP/IP database server jobs running on behalf of the specified local TCP/IP port. Jobs named QRWTSRVR and QZDASOINIT are examples of these server jobs.

***NONE**
Port filtering is not specified.

1-65535
Specify the number of the local port for which monitor records are to be created.

Top

Filter by query governor (FTRQRYGOVR)

Specifies filtering based on the query governor limits.

***NONE**
Query governor filtering is not specified.

***ALL** Monitor records will be collected when a query governor limit is exceeded, regardless if the query is cancelled or not.

***COND**
Monitor records will be collected when a query governor limit is exceeded based on one of the following criteria:

1. Query governor exit program returns 2 indicating that exceeded limit should be ignored, but Database Monitor records should be collected anyway.
2. Query governor exit program returns 3 indicating that the query should be cancelled.
3. Query governor exit program returns 0 indicating that the inquiry message should be issued and the response to the inquiry message is to cancel the query.
4. Query governor exit program fails and the response to the inquiry message is to cancel the query.
5. There is no query governor exit program in effect and the response to the inquiry message is to cancel the query.

   For more information on the query governor exit program, refer to the query governor exit program (QIBM_QQQ_QUERY_GOVR) documentation in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/.

Top

Comment (COMMENT)

User-specified description that is associated with the database monitor. The description is stored in the monitor record that has a record ID of 3018.

***BLANK**
Text is not specified.

character-value
Specify up to 100 characters of text.

Top

Examples

Example 1: Starting Public Monitoring

```
STRDBMON   OUTFILE(QGPL/FILE1)  OUTMBR(MEMBER1 *ADD)
           JOB(*ALL)  FRCRCD(10)
```

This command starts database monitoring for all jobs on the system. The performance statistics are added to the member named MEMBER1 in the file named FILE1 in the QGPL library. Ten records will be held before being written to the file.

Example 2: Starting Private Monitoring

```
STRDBMON   OUTFILE(*LIBL/FILE3)  OUTMBR(MEMBER2)
           JOB(134543/QPGMR/DSP01)  FRCRCD(20)
```

This command starts database monitoring for job number 134543. The job name is DSP01 and was started by the user named QPGMR. The performance statistics are added to the member named MEMBER2 in the file named FILE3. Twenty records will be held before being written to the file.

Example 3: Starting Private Monitoring to a File in a Library in an Independent ASP

```
STRDBMON   OUTFILE(LIB41/DBMONFILE)  JOB(134543/QPGMR/DSP01)
```

This command starts database monitoring for job number 134543. The job name is DSP01 and was started by the user named QPGMR. The performance statistics are added to the member name DBMONFILE (since OUTMBR was not specified) in the file named DBMONFILE in the library named LIB41. This library may exist in more than one independent auxiliary storage pool (ASP); the library in the name space of the originator's job will always be used.

Example 4: Starting Public Monitoring For All Jobs That Begin With 'QZDA'

```
STRDBMON   OUTFILE(LIB41/DBMONFILE)  JOB(*ALL/*ALL/QZDA*)
```

This command starts database monitoring for all jobs that whose job name begins with 'QZDA'. The performance statistics (monitor records) are added to member DBMONFILE (since OUTMBR was not specified) in file DBMONFILE in library LIB41. This library may exist in more than one independent auxiliary storage pool (ASP); the library in the name space of the originator's job will always be used.

Example 5: Starting Public Monitoring and Filtering SQL Statements That Run Over 10 Seconds

```
STRDBMON   OUTFILE(LIB41/DBMONFILE)  JOB(*ALL)  RUNTHLD(10)
```

This command starts database monitoring for all jobs. Monitor records are created only for those SQL statements whose estimated run time meets or exceeds 10 seconds.

Example 6: Starting Public Monitoring and Filtering SQL Statements That Have an Estimated Temporary Storage Over 200 Megabytes

```
STRDBMON   OUTFILE(LIB41/DBMONFILE)  JOB(*ALL)  STGTHLD(200)
```

This command starts database monitoring for all jobs. Monitor records are created only for those SQL statements whose estimated temporary storage meets or exceeds 200 megabytes.

Example 7: Starting Private Monitoring and Filtering Over a Specific File

```
STRDBMON   OUTFILE(LIB41/DBMONFILE) JOB(*)
           FTRFILE(LIB41/TABLE1)
```

This command starts database monitoring for the current job. Monitor records are created only for those SQL statements that use file LIB41/TABLE1.

Example 8: Starting Private Monitoring for the Current User

```
STRDBMON   OUTFILE(LIB41/DBMONFILE)  JOB(*)  FTRUSER(*CURRENT)
```

This command starts database monitoring for the current job. Monitor records are created only for those SQL statements that are executed by the current user.

Example 9: Starting Public Monitoring For Jobs Beginning With 'QZDA' and Filtering Over Run Time and File

```
STRDBMON   OUTFILE(LIB41/DBMONFILE)  JOB(*ALL/*ALL/QZDA*)
           RUNTHLD(10)  FTRUSER(DEVLPR1)  FTRFILE(LIB41/TTT*)
```

This command starts database monitoring for all jobs whose job name begins with 'QZDA'. Monitor records are created only for those SQL statements that meet all of the following conditions:

- The estimated run time, as calculated by the query optimizer, meets or exceeds 10 seconds
- Was executed by user 'DEVLPR1'.
- Use any file whose name begins with 'TTT' and resides in library LIB41.

Example 10: Starting Public Monitoring and Filtering SQL Statements That Have Internet Address 9.10.111.77.

```
STRDBMON   OUTFILE(LIB41/DBMONFILE)  JOB(*ALL)
           FTRINTNETA(9.10.111.77)
```

This command starts database monitoring for all jobs. Monitor records are created only for TCP/IP database server jobs that are using the client IP version 4 address of 9.10.111.77.

Example 11: Starting Public Monitoring and Filtering SQL Statements That Have a Port Number of 8471

```
STRDBMON   OUTFILE(LIB41/DBMONFILE)  JOB(*ALL)  FTRLCLPORT(8471)
```

This command starts database monitoring for all jobs. Monitor records are created only for TCP/IP database server jobs that are using the local port number 8471.

Example 12: Starting Public Monitoring Based on Feedback from the Query Governor

```
CHGSYSVAL QQRYTIMLMT(200)
STRDBMON   OUTFILE(LIB41/DBMONFILE) JOB(*ALL) FTRQRYGOVR(*COND)
```

This commands starts database monitoring for all jobs whose estimated run time is expected to exceed 200 seconds, based on the response to the query governor. In this example data will be collected only if the query is cancelled or a return code of 2 is returned by a query governor exit program. The query can be cancelled by a user response to the inquiry message CPA4259 (that is issued because the query exceeded the query governor limits) or it can be cancelled by the program logic inside the registered query governor exit program.

Top

Error messages

***ESCAPE Messages**

CPF1321
Job &1 user &2 job number &3 not found.

CPF222E
&1 special authority is required.

CPF4269
Not authorized to object &1 in &2 type *&3.

CPF436A
Record format for file &1 in &2 does not match model file.

CPF436B
&1 can not be specified on the OUTFILE parameter.

CPF436C
Job &4 is already being monitored.

CPF436E
Job &1 user &2 job number &3 is not active.

CPF43A2
Address specified on FTRINTNETA parameter is not valid.

***STATUS Messages**

CPI436A
Database monitor started for job &1, monitor ID &2.

Top

Start Data Base Reader (STRDBRDR)

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

Parameters
Examples
Error messages

The Start Database Reader (STRDBRDR) command starts a spooling reader using a database file; the reader reads a batch input stream from the database and places the jobs onto one or more job queues. This command specifies the name of the database file and member from which the input stream is read, the name of the reader, and the names of the job queue and message queue that are used.

More than one reader can be active at the same time (as determined by the spooled subsystem description). Each database reader must have a unique reader name, and the specified file or member must be available. The reader can also be held or canceled by using the Hold Reader (HLDRDR) command or End Reader (ENDRDR) command.

Because each reader runs independently of the job that started it, the user can continue doing other work on the system after he has started a reader.

Restriction: The specified database file either must consist of single-field records and must have an arrival sequence access path, or it must be a standard database source file.

Top

Parameters

Keyword	Description	Choices	Notes
FILE	Data base file	*Qualified object name*	Required, Positional 1
	Qualifier 1: Data base file	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
MBR	Member	*Name*, ***FIRST**	Optional, Positional 2
JOBQ	Job queue	*Qualified object name*	Optional
	Qualifier 1: Job queue	*Name*, **QBATCH**	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
MSGQ	Queue for reader messages	Single values: *REQUESTER Other values: *Qualified object name*	Optional
	Qualifier 1: Queue for reader messages	*Name*, **QSYSOPR**	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
RDR	Reader	*Name*, ***FILE**	Optional

Top

Data base file (FILE)

Specifies the database file from which the input stream is to be read. The file must be available for allocation to the spooling reader before the reader can be started.

This is a required parameter.

Qualifier 1: Data base file

name Specify the name of the file that contains the input stream read by the reader.

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.

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

Top

Member (MBR)

Specifies the member in the specified file that contains the input stream.

***FIRST**
The first member in the file is used.

name Specify the name of the member that contains the input stream to be read.

Top

Job queue (JOBQ)

Specifies the job queue where the spooling reader will place entries. This value is used if *RDR is specified on the **Job queue (JOBQ)** parameter of the Batch Job (BCHJOB) command. (Note that the job queue for each job within this input stream can be different.)

Qualifier 1: Job queue

QBATCH
The job entry is placed on the QBATCH job queue.

name Specify the name of the job queue to be used by this reader.

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 queue.

name Specify the name of the library where the job queue is located.

Top

Queue for reader messages (MSGQ)

Specifies the message queue that will receive any messages that are created by the reader.

Single values

***REQUESTER**
The messages are to be sent to the workstation message queue of the workstation of the user who started this reader. If this value is used in a batch job, the message is changed to the system operator's message queue (QSYSOPR).

Qualifier 1: Queue for reader messages

QSYSOPR
The messages are sent to the system operator's message queue (QSYSOPR).

name Specify the name of the message queue that you want to receive any messages.

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 message queue. If no current library entry exists in the library list, QGPL is used.

name Specify the name of the library where the message queue is located.

Top

Reader (RDR)

Specifies the reader to be started. Each reader name must be unique.

***FILE** The name of the reader is the same as the name of the database file that is specified on the **Data base file (FILE)** parameter.

name Specify the name that identifies the reader to be started.

Top

Examples

```
STRDBRDR   FILE(QGPL/BILLING)
```

This command starts a spooled reader that reads its input from the database file named BILLING, which is in the QGPL library. The reader name is also BILLING because the RDR parameter was not specified. The first member in the BILLING file contains the input stream to be processed. The default job queue QBATCH and the system-supplied system operator's message queue QSYSOPR are used by the database reader.

Top

Error messages

***ESCAPE Messages**

CPF1338
Errors occurred on SBMJOB command.

CPF2207
Not authorized to use object &1 in library &3 type *&2.

CPF3301
Reader &1 already started.

CPF3307
Job queue &1 in &2 not found.

CPF3330
Necessary resource not available.

CPF3362
Objects in QTEMP not valid for parameter values.

CPF3363
Message queue &1 in library &2 not found.

CPF3364
File &1 in library &2 not database file or DDM file.

CPF9812
File &1 in library &2 not found.

CPF9815
Member &5 file &2 in library &3 not found.

Top

Start DFU (STRDFU)

Where allowed to run: Interactive environments (*INTERACT *IPGM *IREXX *EXEC)
Threadsafe: No

Parameters
Examples
Error messages

The Control Language (CL) command STRDFU starts the data file utility (DFU).

Error messages for STRDFU

***ESCAPE Messages**

DFU0005
The command failed.

DFU0018
The Run option is not valid.

Top

Parameters

Keyword	Description	Choices	Notes
OPTION	Option	*Element list*	Optional, Positional 1
	Element 1: DFU option	1-5, ***SELECT**	
	Element 2: Run option	1-2, ***NONE**	
DFUPGM	DFU program	*Qualified object name*	Optional, Positional 2
	Qualifier 1: DFU program	*Name*, ***PRV**	
	Qualifier 2: Library	*Name*, ***PRV**, *LIBL, *CURLIB	
FILE	Data base file	*Qualified object name*	Optional, Positional 3
	Qualifier 1: Data base file	*Name*, ***PRV**, *SAME	
	Qualifier 2: Library	*Name*, ***PRV**, *LIBL, *CURLIB	
MBR	Member	*Name*, ***PRV**, *FIRST	Optional, Positional 4

Top

Option (OPTION)

Specifies the option to use as a value for the DFU main menu.

The possible values are:

***SELECT**
The DFU main menu appears. You can select an option from the menu.

first-menu-option
Type a number between 1 and 5. Type 1 to run, 2 to create, 3 to change, or 4 to delete a DFU program. Type 5 to run a temporary DFU program.

second-menu-option
If you type 1 (to run a DFU program), you can also type a second option. The second option must be 1 or 2. Type 1 to change data, or 2 to display data without changing it.

Top

DFU program (DFUPGM)

Specifies the name of the DFU program to be run, created, changed, or deleted. You do not use this parameter if you select option 5 (to run a temporary DFU program).

The possible values are:

***PRV** DFU will use the program that was used in your last DFU session.

program-name
Type the qualified name of the DFU program to be used.

Top

Data base file (FILE)

Specifies the database file you want to change or display.

The possible values are:

***PRV** DFU will use the file that was used in your last DFU session.

***SAME**
DFU will use the file that was used to define the program. You can specify *SAME only if you select option 1 (to run a DFU program) or option 3 (to change a DFU program).

file-name
Type the qualified name of the data file you want DFU to process. (If you do not specify a library name, *LIBL is used.)

Note: Make sure the data-file name is different from the DFU program name.

Top

Member (MBR)

Specifies the member in the file you want to change or display.

The possible values are:

***PRV** DFU will use the member that was used in your last DFU session.

member-name
Type the name of the member you want DFU to process.

Top

Examples

None

Top

Error messages

***ESCAPE Messages**

DFU0005
The command failed.

DFU0018
The Run option is not valid.

Top

Start DIG Query (STRDIGQRY)

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

Parameters
Examples
Error messages

The Start DIG Query (STRDIGQRY) command, or its alias DIG, starts the Domain Information Groper tool.

DIG is a powerful query tool that allows you to retrieve information from or test the response of a Domain Name System (DNS) server. You can verify that a DNS server is responding correctly before you configure your system to use it. You can also retrieve DNS information about hosts, domains, and other DNS servers.

Unless it is told to query a specific name server, DIG will try each of the servers listed in CHGTCPDMN.

Restrictions:

- You must have execute (*X) authority to the directories in the path of the batch input file.
- You must have read (*R) authority to the batch input file.
- You must have execute (*X) authority to the directories in the path of the key file.
- You must have read (*R) authority to the key file.
- You must have execute (*X) authority to the directories in the path of the trusted keys file.
- You must have read (*R) authority to the trusted keys file.
- You must have execute (*X) authority to the directories in the path of the output file.
- You must have write (*W) authority to the output file if it already exists.
- You must have read, write and execute (*RWX) authority to the output file's parent directory if the output file does not already exist.

Top

Parameters

Keyword	Description	Choices	Notes
HOSTNAME	Query name	*Character value*, ***DFT**	Optional, Positional 1
TYPE	Query type	***A**, *AAAA, *ANY, *AXFR, *CNAME, *MX, *NS, *PTR, *SOA, *SRV, *TXT	Optional, Positional 2
CLASS	Query class	***IN**, *CH, *HS, *ANY	Optional
REVERSE	Reverse lookup	***NO**, *YES, *IP6INT	Optional
DMNNAMSVR	Domain name server	*Character value*, ***CFG**	Optional
PORT	Domain name server port	1-65535, **53**	Optional
TIMEOUT	Query timeout	1-100, **5**	Optional
USEDMNSCHL	Use domain search list	*YES, ***NO**	Optional
DMNSCHLIST	Domain search list	*Character value*, ***CFG**, *NONE	Optional
SRCADR	Source address	*Character value*, ***DFT**, *ANY4, *LOOPBACK4, *ANY6, *LOOPBACK6	Optional
BCHFILE	Batch input file	*Path name*, ***NONE**	Optional
IPVSN	IP Version	***ALL**, *IPV4ONLY, *IPV6ONLY	Optional

Keyword	Description	Choices	Notes
PROTOCOL	Network protocol	***UDP**, *TCP	Optional
SETRDFLAG	Recursion desired	***YES**, *NO	Optional
SETAAFLAG	Authoritative answers only	*Character value*, ***NO**, *YES	Optional
SETADFLAG	Authentic data	***NO**, *YES	Optional
SETCDFLAG	Disable DNSSEC checking	***NO**, *YES	Optional
MULTILINE	Print multiple lines	***NO**, *YES	Optional
SHORT	Print short answer	***NO**, *YES	Optional
IDENTIFY	Print server in short answer	***NO**, *YES	Optional
PRTQRY	Print query	***NO**, *YES	Optional
PRTCLASS	Print RR class	***YES**, *NO	Optional
PRTTTL	Print RR TTL	***YES**, *NO	Optional
PRTALL	Print all query detail	***YES**, *NO	Optional
CMD	Print query command	***DFT**, *YES, *NO	Optional
COMMENTS	Print query comments	***DFT**, *YES, *NO	Optional
STATS	Print query statistics	***DFT**, *YES, *NO	Optional
QUESTION	Print question section	***DFT**, *YES, *NO	Optional
ANSWER	Print answer section	***DFT**, *YES, *NO	Optional
AUTHORITY	Print authority section	***DFT**, *YES, *NO	Optional
ADDITIONAL	Print additional section	***DFT**, *YES, *NO	Optional
KEYFILE	Key file	*Path name*, ***NONE**	Optional
KEYNAME	Key name	*Character value*, ***NONE**	Optional
STOPFAIL	Stop on SERVFAIL	***YES**, *NO	Optional
UDPTRUNC	Ignore truncated responses	***RETRY**, *IGNORE	Optional
NSSCH	List authoritative servers	***NO**, *YES	Optional
TRACE	Trace delegation path	***NO**, *YES	Optional
UDPTRIES	Times to try UDP query	1-100, **3**	Optional
UDPNBRRTY	UDP retry	0-100, **2**	Optional
NBRDOTS	Number of dots	0-10, **1**	Optional
BUFSIZE	UDP buffer size for EDNS	0-65535, **0**	Optional
EDNS	Set EDNS version	0-255, **0**	Optional
NOEDNS	Clear EDNS version	***NO**, *YES	Optional
BESTEFFORT	Best effort display	***NO**, *YES	Optional
DNSSEC	Request DNSSEC records	***NO**, *YES	Optional
SIGCHASE	Chase DNSSEC chains	***NO**, *YES	Optional
TRUSTEDKEY	Trusted keys file	*Path name*, ***NONE**	Optional
TOPDOWN	DNSSEC top down validation	***NO**, *YES	Optional
TOSTMF	Output file	*Path name*, ***STDOUT**	Optional

Top

Query name (HOSTNAME)

Specifies the name that you want the Domain Name System (DNS) server to supply information about. You can use either a name or an IP address to identify the resource record. For example, a name could be 'mycomputer.mycompany.com' An IP address could be an IPv4 address like '10.0.1.100' or an IPv6 address like '2001:D88::1'. You can use either the name to obtain the address, or the address to obtain the name.

***DFT** Query the root '.' servers.

domain-name
Specify the name to use for the DNS server query. This is usually a hostname (like mycomputer.mycompany.com) or domain (like mycompany.com) that you want to query.

internet-address
Specify the IPv4 address in dotted-decimal notation or a colon-delimited IPv6 address. If you use an IP address, consider using the REVERSE(*YES) parameter to simplify PTR (pointer) type queries. See the examples for this command for more information.

Top

Query type (TYPE)

Specifies the type of the query. The default type is *A. The list of supported query types changes with time, and not all servers support all the types that exist. This list provides some common query types for convenience and is not a complete list. If you do not see a query type you want in this list, you can still type in a character string that represents that query type. If the type is unknown by this BIND version, the query will default to an *A query with any corresponding results.

***A** IPv4 Address record. This is the character string 'a'.

***AAAA**
IPv6 address record. This is the character string 'aaaa'.

***ANY** Any resource record. This is the character string 'any'.

***AXFR**
Zone transfer. This is the character string 'axfr'.

***CNAME**
Canonical name record. Returns a list of aliases for the true (canonical) host name, if any exist. This is the character string 'cname'.

***MX** Mail exchange record. This is the character string 'mx'.

***NS** Name server (DNS server) information for the zone. This is the character string 'ns'.

***PTR** Pointer record. Returns a name for an IP address. This is the character string 'ptr'.

***SOA** Start of authority record. This is the character string 'soa'.

***SRV** Services location selection. This is the character string 'srv'.

***TXT** Text record. This is the character string 'txt'.

Top

Query class (CLASS)

Specifies the protocol group of the information.

***IN** The Internet class.

***CH** The CHAOS class. The server provides some helpful diagnostic information through a number of built-in zones under the CHAOS class.

version

If your BIND server named.conf has the ’version’ option configured, then you can query for it using the CHAOS class. This is the version the server should report via a query of the name ’version.bind’ with type TXT, class CHAOS. The default is the real version number of this server.

```
DIG   HOSTNAME('version.bind')  TYPE(*TXT)  CLASS(*CH)
```

hostname

If your BIND server named.conf has the ’hostname’ option configured, then you can query for it using the CHAOS class. This is the hostname the server should report via a query of the name ’hostname.bind’ with type TXT, class CHAOS. This defaults to the hostname of the machine hosting the name server. The primary purpose of such queries is to identify which of a group of anycast servers is actually answering your queries.

```
DIG   HOSTNAME('hostname.bind')  TYPE(*TXT)  CLASS(*CH)
```

server-id

If your BIND server named.conf has the ’server-id’ option configured, then you can query for it using the CHAOS class. This is the ID of the server should report via a query of the name ’ID.SERVER’ with type TXT, class CHAOS. The primary purpose of such queries is to identify which of a group of anycast servers is actually answering your queries. The default server-id is none.

```
DIG   HOSTNAME('ID.SERVER')  TYPE(*TXT)  CLASS(*CH)
```

***HS** Specifies the MIT Athena Hesiod class. Hesiod, developed by MIT Project Athena, is an information service built upon BIND. Its intent is to furnish information about users, groups, network-accessible file systems, printcaps, and mail service throughout an installation. In other words, it holds arbitrary data stored as text strings. Queries that use the HS class retrieve data stored as TXT Resource Records. You cannot specify root servers for the Hesiod class (generally the ones at MIT) in your list of root name servers.

***ANY** This says to query any class, and is a wildcard query.

Top

Reverse lookup (REVERSE)

Specifies if the value specified for the **Query name (HOSTNAME)** parameter is to be used as part of a reverse (PTR) query. A reverse lookup is when you map an IPv4 address in dotted-decimal notation or a colon-delimited IPv6 address to a name. Reverse lookups require the name to be in either the ’in-addr.arpa’ (IPv4) or ’ip6.arpa’ (IPv6) zone.

Reverse lookups are simplified by this option as there is no need format the NAME parameter or provide the TYPE and CLASS parameters. For an IPv4 address like 10.0.1.100, DIG will reverse the IP address using byte (8-bit) format and sets the NAME parameter to 100.1.0.10.in-addr.arpa. DIG also sets the TYPE and CLASS parameters to *PTR and *IN, respectively.

By default, IPv6 addresses are queried using nibble (4-bit) format under the zone ip6.arpa (and not zone ip6.int). Thus, for an IPv6 address like 2001:D88::1, DIG will set the NAME parameter to 1.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.8.8.d.0.1.0.0.2.ip6.arpa. DIG also sets the TYPE and CLASS parameters to *PTR and *IN, respectively.

Note: For IPv4, a reverse lookup name is normally the IP address written in reverse order, separated by a dot at every 8-bit boundary (a ’byte’ boundary) and it is delegated within the zone in-addr.arpa (i.e.

100.1.0.10.in-addr.arpa.). The notation for IPv6 is a bit different, it is also the IP address written backwards with dots as separators, but it is separated at every 4-bit boundary (a 'nibble' boundary) and the delegation is done within the zone ip6.arpa. (e.g. 8.8.d.0.1.0.0.2.ip6.arpa.). In the past, the delegation for IPv6 addresses was done in the zone ip6.int, therefore today it might still be common and often necessary to query a delegation like 8.8.d.0.1.0.0.2.ip6.int.

***NO** No reverse lookup will be performed.

***YES** A reverse lookup will be performed.

***IP6INT**
An IPv6 address reverse lookup in the zone ip6.int will be performed. This zone is deprecated, but may still be required to query IPv6 backbone prefixes.

Top

Domain name server (DMNNAMSVR)

Specifies the name or the IP address of the DNS server that DIG will use as its current server for the query session. You can specify any DNS server to which your TCP/IP network has access.

DIG retrieves information from DNS servers. It needs an active DNS server to send its queries to. If you do not specify a DNS server with DMNNAMSVR when you start the tool, it will attempt to set one of the following as its default DNS server for the session:

1. DNS server your system is configured to use (*CFG), or
2. The DNS server that is running on your local system.

***CFG** Try all the DNS servers currently designated for use by this system. These server internet addresses can be seen by prompting the Change TCP/IP Domain (CHGTCPDMN) command and looking at the values shown for the INTNETADR parameter.

server-domain-name
Specify the name of a DNS server.

server-internet-address
Specify the IP address of a DNS server.

Top

Domain name server port (PORT)

Specifies the default TCP/UDP name server port to use.

53 The default port is 53.

1-65535
Specify a valid port number.

Top

Query timeout (TIMEOUT)

Specifies the timeout interval, in seconds, to wait for a response.

5 A timeout value of 5 seconds is used.

1-100 Specify a valid timeout value in seconds.

Top

Use domain search list (USEDMNSCHL)

Specifies whether or not to use the search list defined by the **Domain search list (DMNSCHLIST)** parameter. The search list is not used by default.

_*NO_ Do not use a search list.

***YES** Use the search list specified by the DMNSCHLIST parameter.

Top

Domain search list (DMNSCHLIST)

Specifies a single domain name for the search list. The default search list is set by running the Change TCP/IP Domain (CHGTCPDMN) command and specifying the DMNSCHLIST parameter. If more than one domain name is found, only the first domain name is used.

If DMNSCHLIST is specified, USEDMNSCHL(*YES) must also be specified.

*CFG Use the first domain name found in the search list specified for the DMNSCHLIST parameter on the CHGTCPDMN command.

***NONE**
Do not specify a search list.

character-value
Specify a valid domain name string to use. Only one domain name can be specified.

Top

Source address (SRCADR)

Specifies the source IP address to use when sending the query. Some DNS servers are configured to only allow queries from certain source addresses. This parameter allows you use a specific source address if your system has multiple network interfaces.

*DFT Uses the default supplied by the stack.

***ANY4**
Any IPv4 address, or ’0.0.0.0’.

***LOOPBACK4**
IPv4 loopback, or ’127.0.0.1’.

***ANY6**
Any IPv6 address, or ’::’.

***LOOPBACK6**
IPv6 loopback, or ’::1’.

character-value
Specify an IP address to use as the source IP address for the query.

Top

Batch input file (BCHFILE)

Specifies the file containing a list of query requests to be used as batch input. The file contains a number of queries, one per line. Each entry in the file should be organized in the same way they would be presented as queries to DIG using the PASE command-line interface. The purpose of using a file is to allow batch mode or scripting operation.

***NONE**
Do not specify a batch query file.

path-name
Specify the path for a stream file from which input is read. The contents of the stream file lines are as follows:

```
Format: [@global-server] [domain] [q-type] [q-class] {q-opt}
                or
        {global-d-opt} domain [@local-server] {local-d-opt}
                or
        [domain [@local-server] {local-d-opt} [...]]
Where:
 domain   is in the Domain Name System
 q-type   is one of (a,any,mx,ns,soa,...) [default:a]
          (Use ixfr=version for type ixfr)
 q-class  is one of (in,hs,ch, ...) [default: in]
 q-opt    is one of:
          -x dot-notation     (shortcut for in-addr lookups)
          -i                  (IP6.INT reverse IPv6 lookups)
          -f filename         (batch mode)
          -b address[#port]   (bind to source address/port)
          -p port             (specify port number)
          -q name             (specify query name)
          -t type             (specify query type)
          -c class            (specify query class)
          -k keyfile          (specify tsig key file)
          -y [hmac:]name:key  (specify named base64 tsig key)
          -4                  (use IPv4 query transport only)
          -6                  (use IPv6 query transport only)
 d-opt  is of the form +keyword[=value], where keyword is:
          +[no]vc             (TCP mode)
          +[no]tcp            (TCP mode, alternate syntax)
          +time=###           (Set query timeout) [5]
          +tries=###          (Set number of UDP attempts) [3]
          +retry=###          (Set number of UDP retries) [2]
          +domain=###         (Set default domainname)
          +bufsize=###        (Set EDNS0 Max UDP packet size)
          +ndots=###          (Set NDOTS value)
          +edns=###           (Set EDNS version)
          +[no]search         (Set whether to use searchlist)
          +[no]showsearch     (Search with intermediate results)
          +[no]defname        (Ditto)
          +[no]recurse        (Recursive mode)
          +[no]all            (Set or clear all display flags)
          +[no]qr             (Print question before sending)
          +[no]nssearch       (Search authoritative nameservers)
          +[no]identify       (ID responders in short answers)
          +[no]trace          (Trace delegation down from root)
          +[no]dnssec         (Request DNSSEC records)
          +[no]sigchase       (Chase DNSSEC signatures)
          +trusted-key=####   (Trusted Key chasing DNSSEC sigs)
          +[no]topdown        (Do DNSSEC validation top down)
          +[no]multiline      (Print records in expanded format)
 global d-opts and servers (before domain) affect all queries
 local d-opts and servers (after domain) affect only that query
For example, you could put these lines into a file:
  ibm.com aaaa in
```

```
aol.com mx
-q microsoft.com -t aaaa -c in
+recurse cisco.com @10.0.0.1 @10.0.0.2
@10.0.0.1 aa.com
```

Top

IP Version (IPVSN)

Specifies whether to limit the query to IPv4 or IPv6 networks.

*ALL
Do not limit queries to IPv4 or IPv6.

*IPV4ONLY
Only send queries out IPv4 network interfaces.

*IPV6ONLY
Only send queries out IPv6 network interfaces.

Top

Network protocol (PROTOCOL)

Specified whether to use TCP or UDP when sending requests to the server.

*UDP
Use UDP to send the query. However, TCP will be automatically selected for queries that require it, such as zone transfer (AXFR) requests.

*TCP
Use TCP to send the query.

Top

Recursion desired (SETRDFLAG)

Specifies whether or not to set the Recursion Desired (RD) flag in the query. This tells the current DNS server to query other DNS servers if the current server does not have the information.

*YES
Set the RD flag.

*NO
Do not set the RD flag.

Top

Authoritative answers only (SETAAFLAG)

Specifies whether or not to set the Authoritative Answers (AA) flag in the query. This indicates that you want the response to be from an authoritative server and not from the cache of a non-authoritative server. Authoritative servers own the domain being queried.

*NO
Do not set the AA flag.

*YES
Set the AA flag.

Top

Authentic data (SETADFLAG)

Specifies whether or not to set the Authentic Data (AD) flag in the query. The AD bit currently has a standard meaning only in responses, not in queries, but the ability to set the bit in the query is provided for completeness.

***NO** Do not set the AD flag.

***YES** Set the AD flag.

Top

Disable DNSSEC checking (SETCDFLAG)

Specifies whether or not to set the Checking Disabled (CD) flag in the query. This requests the server to not perform DNSSEC validation of responses.

***NO** Do not set the CD flag.

***YES** Set the CD flag.

Top

Print multiple lines (MULTILINE)

Specifies whether or not to print records, like the SOA record, in a verbose multi-line format with human-readable comments.

***NO** Do not use multi-line format. Print each record on a single line, which can facilitate machine parsing of the output.

***YES** Use multi-line format.

Top

Print short answer (SHORT)

Specifies whether or not DIG should provide a terse answer.

***NO** Do not print a terse answer. Print the answer in a verbose form.

***YES** Print a terse answer.

Top

Print server in short answer (IDENTIFY)

Specifies whether or not to display the IP address and port number of the server that supplied the answer when the SHORT(*YES) parameter is specified.

Note: This parameter is ignored if SHORT(*NO) is specified.

***NO** Do not print the IP address and port number of the server providing the answer.

***YES** Print the IP address and port number of the server providing the answer.

Top

Print query (PRTQRY)

Specifies whether or not to print the query as it is sent.

***NO** Do not print the query as it is sent.

***YES** Print the query as it is sent.

Top

Print RR class (PRTCLASS)

Specifies whether or not to print the CLASS when printing the record.

***YES** Print the CLASS in the record output.

***NO** Do not print the CLASS in the record output.

Top

Print RR TTL (PRTTTL)

Specifies whether or not to print the Time To Live (TTL) in the record output.

***YES** Print the TTL in the record output.

***NO** Do not print the TTL in the record output.

Top

Print all query detail (PRTALL)

Specifies whether to set or clear all print flags. You use this parameter in conjunction with other print parameters. This allows you to print only the output you are interested in. The shipped default is *YES for the CMD, COMMENTS, STATS, QUESTION, ANSWER, AUTHORITY and ADDITIONAL parameters.

***YES** Set all print flags.

***NO** Clear all print flags.

For example, you could use the PRTALL parameter as follows:

```
DIG   PRTALL(*NO)  QUESTION(*YES)  ANSWER(*YES)
```

The PRTALL(*NO) parameter turns off all print flags, and the QUESTION(*YES) and ANSWER(*YES) flags override the PRTALL(*NO) parameter to print only the question and answer sections of the DIG output.

Top

Print query command (CMD)

Specifies whether to print the initial comment in the output identifying the version of DIG and the query options that have been applied.

***DFT** Use the value specified for the **Print all query detail (PRTALL)** parameter.

***YES** Print the initial comment.

***NO** Do not print the initial comment.

Top

Print query comments (COMMENTS)

Specifies whether to print comment lines in the output.

***DFT** Use the value specified for the **Print all query detail (PRTALL)** parameter.

***YES** Print the comment lines.

***NO** Do not print the comment lines.

Top

Print query statistics (STATS)

Specifies whether to print query statistics, like when the query was made or the size of the reply.

***DFT** Use the value specified for the **Print all query detail (PRTALL)** parameter.

***YES** Print the query statistics.

***NO** Do not print the query statistics.

Top

Print question section (QUESTION)

Specifies whether to print the question section of a reply.

***DFT** Use the value specified for the **Print all query detail (PRTALL)** parameter.

***YES** Print the question section of a reply.

***NO** Do not print the question section of a reply.

Top

Print answer section (ANSWER)

Specifies whether to print the answer section of a reply.

***DFT** Use the value specified for the **Print all query detail (PRTALL)** parameter.

***YES** Print the answer section of a reply.

***NO** Do not print the answer section of a reply.

Top

Print authority section (AUTHORITY)

Specifies whether to print the authority section of a reply.

***DFT** Use the value specified for the **Print all query detail (PRTALL)** parameter.

***YES** Print the authority section of a reply.

***NO** Do not print the authority section of a reply.

Top

Print additional section (ADDITIONAL)

Specifies whether to print the additional section of a reply.

***DFT** Use the value specified for the **Print all query detail (PRTALL)** parameter.

***YES** Print the additional section of a reply.

***NO** Do not print the additional section of a reply.

Top

Key file (KEYFILE)

Specifies a Transaction Signature (TSIG) key file to sign the DNS queries. The DNS server that is queried needs to be configured with the TSIG key and algorithm that is being used or the transaction will fail.

This key must be a base-64 encoding of an HMAC-MD5 key.

Note: If this parameter is specified, the KEYNAME parameter cannot be specified.

***NONE**
Do not specify a key path name.

path-name
Specify the path name of a stream file which contains the keys to be used. For example, '/QIBM/UserData/OS400/DNS/_DYN/my-tsig-key._KID'.

The DNS server being queried needs to include this key name and algorithm in its configuration in order to accept this TSIG key from clients.

Top

Key name (KEYNAME)

Specifies a Transaction Signature (TSIG) key to sign the DNS queries. The key is a base-64 encoded string. The DNS server that is queried needs to be configured with the TSIG key and algorithm that is being used or the transaction will fail.

Note: If this parameter is specified, the KEYFILE parameter cannot be specified.

***NONE**
Do not specify a key name.

character-value
Specify the Transaction Signature key to use. The format is *name:key* where *name* is the key name, and *key* is the actual key as a base-64 encoded string. Example:

```
my-tsig-key:JNvcpnxysbJ2hsdOqQ5qrQ==
```

The key name in this case is 'my-tsig-key' and the base-64 encoded key is 'JNvcpnxysbJ2hsdOqQ5qrQ=='.

The DNS server being queried needs to include this key and algorithm in its configuration in order to accept this TSIG key from clients.

Top

Stop on SERVFAIL (STOPFAIL)

Specifies whether to stop or try the next server if you receive a SERVFAIL response. The default is to not try the next server, which is the reverse of normal stub resolver behaviour.

***YES** Do not try the next server if a SERVFAIL response is received. This is the reverse of normal stub resolver behaviour.

***NO** Try the next server if a SERVFAIL response is received.

Top

Ignore truncated responses (UDPTRUNC)

Specifies whether to ignore truncation in UDP responses or retry with TCP.

***RETRY**
Retry queries using TCP if UDP responses are truncated.

***IGNORE**
Ignore truncation in UDP responses. Do not retry queries using TCP if UDP responses are truncated.

Top

List authoritative servers (NSSCH)

Specifies whether or not DIG should attempt to find the authoritative DNS servers for the zone containing the name being looked up (HOSTNAME parameter) and print the Start of Authority (SOA) record that each name server has for the zone.

***NO** Do not search for the SOA records of all DNS servers for the zone containing the HOSTNAME.

***YES** Search for the SOA records of all DNS servers for the zone containing the HOSTNAME.

Top

Trace delegation path (TRACE)

Specifies whether or not to tracing the delegation path from the root name servers for the name being looked up. When tracing is enabled, DIG makes iterative queries to resolve the name being looked up. It will follow referrals from the root servers, showing the answer from each server that was used to resolve the lookup. You must have access to the root servers for this option.

***NO** Do not trace the delegation path from the root servers for the name being looked up (HOSTNAME parameter).

***YES** Trace the delegation path from the root servers for the name being looked up (HOSTNAME parameter).

Top

Times to try UDP query (UDPTRIES)

Specifies how many times to try UDP queries to the current DNS server before attempting TCP queries.

3 Try UDP queries three times before attempting TCP queries.

1-100 Specify the number of times to try UDP queries before attempting TCP queries. If you use this parameter, you do not need to use the UDPNBRRTY parameter.

Top

UDP retry (UDPNBRRTY)

Specifies the number of times to retry UDP queries to the current DNS server before attempting TCP queries.

2 Two UDP retries will be used. This is in addition to the initial query.

0-100 Specify the number of UDP retries. You normally only need to retry a query 3 to 5 times. If you use this parameter, you do not need to use the UDPTRIES parameter.

Top

Number of dots (NBRDOTS)

Specifies the number of dots (period characters) that have to appear in a domain name for it to be considered absolute.

Names with fewer dots are interpreted as relative names and will be searched for in the domains listed in the DMNSCHLIST parameter.

1 If the domain name contains one dot it is considered an absolute name.

0-10 Specify the number of dots that must be in a domain name for the domain name to be considered to be an absolute name.

Top

UDP buffer size for EDNS (BUFSIZE)

Specifies the UDP message buffer size advertised using EDNS(0).

0 The UDP message buffer size is 0.

0-65535
Specify the number of bytes to use as the UDP message buffer size. You typically want to set this buffer size to avoid truncated packets in UDP. By default, UDP sends a maximum packet size of 512 bytes unless overridden by this option. This is useful if TCP connections are blocked by firewalls.

Top

Set EDNS version (EDNS)

Specifies the EDNS version to use with queries. Setting the EDNS version will cause an EDNS query to be sent. The **Clear EDNS version (NOEDNS)** parameter clears the remembered EDNS version.

If EDNS is specified, the NOEDNS parameter must not be specified.

0 Version 0 is traditionally used to allow UDP packets larger than 512 bytes for clients and servers that support it.

0-255 Values other than 0 have uses in DNSSEC zones.

Resolvers and servers can negotiate with other servers to set a UDP message size larger than the 512 byte default. Typically, they will attempt to negotiate a 4096 byte UDP packet size with a remote server. This negotiation requires the remote server to support EDNS. If the remote server does not support EDNS, then the original 512 byte limit remains the UDP default.

You can use the BUFSIZE parameter to control the size of the negotiated UDP packets.

Top

Clear EDNS version (NOEDNS)

Specifies that the value set by a previous DIG invocation with the **Set EDNS version (EDNS)** parameter specified is to be cleared.

If NOEDNS is specified, the EDNS parameter must not be specified.

NO Do not clear the remembered EDNS version.

YES Clear the remembered EDNS version.

Top

Best effort display (BESTEFFORT)

Specifies whether or not to attempt to print the contents of messages which are malformed.

NO Do not print malformed answers.

YES Attempt to print malformed answers.

Top

Request DNSSEC records (DNSSEC)

Specifies whether or not to request DNSSEC records be sent by setting the DNSSEC OK bit (DO) in the OPT record in the additional section of the query.

NO Do not request any DNSSEC records.

YES Request any DNSSEC records.

Top

Chase DNSSEC chains (SIGCHASE)

Specifies whether or not to chase DNSSEC signature chains. This attempts to verify SIG records that belong to the record and further will try to verify them recursively for all the keys and DS that form the chain of trust all the way up to any self-signed or unsigned key.

NO Do not chase DNSSEC signature chains.

YES Chase DNSSEC signature chains.

If SIGCHASE(*YES) is specified, DNSSEC(*YES) must also be specified.

Top

Trusted keys file (TRUSTEDKEY)

Specifies a file containing trusted keys to be used with the SIGCHASE parameter. Each DNSKEY record must be on its own line.

If a value other than *NONE is specified, SIGCHASE(*YES) must also be specified.

***NONE**
Do not specify a trusted key.

path-name
Specify the path name for a stream file with the trusted keys.

Top

DNSSEC top down validation (TOPDOWN)

Specifies, when chasing DNSSEC signature chains, whether or not to perform a top down validation (from the root ’.’ to the leaf nodes).

If TOPDOWN(*YES) is specified, SIGCHASE(*YES) must also be specified.

***NO** Do not do top down validation.

***YES** Do top down validation.

Top

Output file (TOSTMF)

Specifies the name of a stream file where all command output is written.

***STDOUT**
All command output goes to the standard output device (normally the display).

path-name
Specify the path for a stream file where output should be written.

Top

Examples

Example 1: Forward Lookup Example - Long

```
STRDIGQRY   HOSTNAME('ibm.com')
            DMNNAMSVR(*CFG)
```

This command illustrates a simple query for a domain named ’ibm.com’. It will use the configured DNS server on the system. By default it will look up record type *A and class *IN. The output from this command might look like this:

```
; <<>> DiG 9.4.1.i5/OS.V6R1M0 <<>> ibm.com @10.0.1.100
; (1 server found)
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 510
;; flags: qr rd ra; QUERY: 1, ANSWER: 3, AUTHORITY: 3, ADDITIONAL: 3
;; QUESTION SECTION:
```

```
;ibm.com.                       IN      A
;; ANSWER SECTION:
ibm.com.                9559    IN      A       129.42.18.103
ibm.com.                9559    IN      A       129.42.16.103
ibm.com.                9559    IN      A       129.42.17.103
;; AUTHORITY SECTION:
ibm.com.                9559    IN      NS      leda2.ibm.com.
ibm.com.                9559    IN      NS      castor.ibm.com.
ibm.com.                9559    IN      NS      pollux.ibm.com.
;; ADDITIONAL SECTION:
leda2.ibm.com.          9559    IN      A       9.14.1.3
castor.ibm.com.         9559    IN      A       9.78.1.2
pollux.ibm.com.         9559    IN      A       9.46.1.2
;; Query time: 13 msec
;; SERVER: 10.0.1.100#53(10.0.1.100)
;; WHEN: Sat Aug  5 11:11:18 2006
;; MSG SIZE  rcvd: 183
```

Example 2: Forward Lookup Example - Short

```
DIG   HOSTNAME('ibm.com')
      DMNNAMSVR(*CFG)
      SHORT(*YES)
      IDENTIFY(*YES)
```

This command illustrates a the same query as in example 1, but with shorter output.

```
129.42.18.103 from server 10.0.1.100 in 1 ms.
129.42.16.103 from server 10.0.1.100 in 2 ms.
129.42.17.103 from server 10.0.1.100 in 2 ms.
```

Example 3: Reverse Lookup Example

```
DIG   HOSTNAME('10.0.1.100')
      REVERSE(*YES)
```

This command illustrates a simple reverse query for a host with IPv4 address '10.0.1.100'. It will use the configured DNS server on the system (seen with the CHGTCPDMN command). By default it will look up name '100.1.0.10.in-addr.arpa.' and record type PTR and class *IN. This is equivalent to the following command:

```
DIG   HOSTNAME('100.1.0.10.in-addr.arpa')
      TYPE(*PTR)
      CLASS(*IN)
```

The output from this type of query might look like this:

```
; <<>> DiG 9.4.1.i5/OS.V6R1M0 <<>> -x 10.0.1.100 @10.0.1.100
; (1 server found)
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 609
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 1
;; QUESTION SECTION:
;100.1.0.10.in-addr.arpa.         IN      PTR
;; ANSWER SECTION:
```

```
100.1.0.10.in-addr.arpa.    86400 IN      PTR     mydns1.i5os.ibm.com.
;; AUTHORITY SECTION:
0.10.in-addr.arpa.          86400 IN      NS      mydns1.i5os.ibm.com.
;; ADDITIONAL SECTION:
mydns2.i5os.ibm.com.        47409 IN      A       10.0.2.200
;; Query time: 994 msec
;; SERVER: 10.0.1.100#53(10.0.1.100)
;; WHEN: Sat Aug  5 11:32:16 2006
;; MSG SIZE  rcvd: 119
```

Example 4: DIG with Searchlist

```
STRDIGQRY   HOSTNAME(time)
            TYPE(*ANY)
            CLASS(*IN)
            USESCHDMNL(*YES)
            DMNSCHLIST(nist.gov)
```

This command queries for any records associated with the name ’time’. If this query fails, then DIG will retry the query by appending the domain ’ibm.com’ from the searchlist to the name.

The output from this type of query might look like this:

```
; <<>> DiG 9.4.1.i5/OS.V6R1M0 <<>> -q time -t any -c in @10.0.1.100
  +search +domain=nist.gov
; (1 server found)
;; global options:  printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 1336
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 2, ADDITIONAL: 2
;; QUESTION SECTION:
;time.nist.gov.                 IN      ANY
;; ANSWER SECTION:
time.nist.gov.          287     IN      A    192.43.244.18
;; AUTHORITY SECTION:
nist.gov.               65982   IN      NS   ns1.nist.gov.
nist.gov.               65982   IN      NS   dns-x.boulder.nist.gov.
;; ADDITIONAL SECTION:
ns1.nist.gov.           283     IN      A    129.6.13.2
dns-x.boulder.nist.gov. 40652   IN      A    132.163.4.9
;; Query time: 1038 msec
;; SERVER: 10.0.1.100#53(10.0.1.100)
;; WHEN: Sat Aug  5 11:36:18 2006
;; MSG SIZE  rcvd: 125
```

Example 5: Suppression of Response Sections

```
DIG   HOSTNAME(time.nist.gov)
      TYPE(*ANY)
      CLASS(*IN)
      PRTALL(*NO)
      QUESTION(*YES)
      ANSWER(*YES)
```

This command is similar to Example 4, except we suppressed most of the response sections using the PRTALL(*NO) paramter, and then overrode that setting for the question section using QUESTION(*YES) and the answer section using ANSWER(*YES).

The output from this type of query might look like this:

```
;time.nist.gov.                 IN      ANY
time.nist.gov.          673     IN      A       192.43.244.18
```

Top

Error messages

***ESCAPE Messages**

DNS0013
Error processing command parameters.

DNS0065
Option 33 of i5/OS is required, but is not installed.

TCP7124
Program &1 in library &2 type *PGM ended abnormally.

Top

Start Directory Shadowing (STRDIRSHD)

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

Parameters
Examples
Error messages

The Start Directory Shadowing (STRDIRSHD) command submits a job to start the directory shadowing environment in the system work subsystem (QSYSWRK). The system administrator can use this command to restart the directory shadowing environment if it is not already active. Only one active directory shadowing environment per system is allowed. If the directory shadowing environment is already active, a warning message is issued.

The system work subsystem (QSYSWRK) automatically starts the directory shadowing environment as a prestart job when the subsystem is started.

To ensure the job submitted with this command is successful, use the Work with Jobs (WRKJOB) command using the job number returned in the message after issuing the STRDIRSHD command.

There are no parameters for this command.

Restriction: You must have job control (*JOBCTL) authority to use this command.

Top

Parameters

None

Top

Examples

```
STRDIRSHD
```

This command submits a job to start the directory shadowing environment in the system work subsystem QSYSWRK.

Top

Error messages

***ESCAPE Messages**

CPF89A8
Unable to start job that controls directory shadowing.

Top

Start NSLOOKUP Query (STRDNSQRY)

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

Parameters
Examples
Error messages

The Start NSLOOKUP Query (STRDNSQRY) command, or its alias NSLOOKUP, starts the Name Server Lookup tool. This tool is deprecated and is only provided for backwards compatibility. The recommended tool is now DIG which can be run using the Start DIG Query (STRDIGQRY or DIG) command. In addition, another simpler tool, HOST, is also recommended which can be run using the Start HOST Query (STRHOSTQRY or HOST) command.

NSLOOKUP is an interactive query tool that allows you to retrieve information from or test the response of a Domain Name System (DNS) server. You can verify that a DNS server is responding correctly before you configure your system to use it. You can also retrieve DNS information about hosts, domains, and other DNS servers.

NSLOOKUP asks for (queries) information from DNS servers. To begin a NSLOOKUP query session, an active DNS server must be designated the ’current’ server for the query session. The current server is the DNS server that NSLOOKUP sends all queries to unless you tell it otherwise. All references in the following help to ’the current server’, or ’the current DNS server’, refer only to the current DNS server for the current NSLOOKUP query session.

NSLOOKUP retrieves information from DNS servers. It needs an active DNS server to which it can send its queries. If you do not specify a DNS server with DMNNAMSVR when you start the tool, it will attempt to set one of the following as its current DNS server for the session:

1. The DNS server your system is configured to use
2. The DNS server that is running on your local system.

If neither of these conditions exist, NSLOOKUP will not be able to retrieve any information until you specify a DNS server to query. DMNNAMSVR allows you to start the query session and set the DNS server of your choice as the default server for the session.

Following is a list of NSLOOKUP subcommands that can be used once the query session is started.

host Look up information for host and query the current DNS server.

host dns-server
Look up information for host, but query dns-server instead of the current DNS server.

Allows you to direct the query to a DNS server other than the current DNS server for the query session.

server dns-server
Change the current DNS server to dns-server, using the current DNS server to get the IP address of dns-server.

lserver dns-server
Change the current DNS server to dns-server, using the initial DNS server to get the IP address of dns-server.

Useful if you switched default DNS servers during your query session, and the current DNS server cannot resolve the new DNS server name. lserver allows you to make the switch using your initial DNS server instead of the current one. If the initial DNS server also cannot resolve the new DNS name, substitute the IP address for the name, if you know it. If you do not know the IP address for the new DNS server, try restarting the NSLOOKUP session using the DMNNAMSVR parameter to specify the new DNS server as the current server for the query session.

root This interactive command is not implemented.

finger This interactive command is not implemented.

ls This interactive command is not implemented.

view This interactive command is not implemented.

help This interactive command is not implemented.

? This interactive command is not implemented.

exit Exits interactive mode.

set keyword

set keyword=value
This command is used to change state information that affects the lookups.

set all Displays the current values of the frequently used options to set. Information about the current DNS server and host is also displayed.

set class=value
Change the default query class to one of the following values:

- IN (the Internet class)
- CH (the CHAOS class)
- HS (the Hesiod class)
- ANY (wildcard - any class)

The class specifies the protocol group of the information. The initial default query class is IN.

set debug
Turn debugging mode on. A lot more information is displayed about the packet sent to the current DNS server and the resulting answer.

set nodebug
Turn debugging mode off.

set d2 Turn exhaustive (verbose) debugging mode on. A lot more information is displayed about the packet sent to the current DNS server and the resulting answer.

set nod2
Turn exhaustive (verbose) debugging mode off.

set domain=name
Set default domain to name.

set search
Append the name in the domain search list to the request.

set nosearch
Do not append the name in the domain search list to the request.

set port=X
Use port ’X’ to query the current DNS server, where ’X’ is a numerical port number. The default value is 53.

Note: The well-known port number for DNS servers is 53 and most DNS servers use it. You do not normally need to set the port value unless the DNS server you want to query is not using port 53. Other ports are sometimes used under special circumstances.

set querytype=X

set type=X
Sets the record type that used in the query. Replace ′X′ with one of the DNS record types (A, CNAME, MX, PTR, AAAA, etc.).

set recurse
Tell the current DNS server to query other DNS servers if the current server does not have the information.

set norecurse
Tell the current DNS server to not query other DNS servers if the current server does not have the information.

set retry=X
Set the number of retries to X (where X is a numerical value). The default value for number of retries is 1.

Note: The retry value works together with the timeout value, which is the time in seconds that NSLOOKUP waits before making the first retry. Retry values are usually set to 1 or 2.

set timeout=X
Set the timeout interval to X seconds (where X is a numerical value). The default value is 5 seconds.

Note: Timeout determines how long NSLOOKUP waits before making the first retry if no reply is received on the first query. The timeout value doubles after each unsuccessful retry.

set vc Use TCP for queries instead of UDP.

set novc
Use UDP for queries instead of TCP.

Restrictions:

- This command is not threadsafe when run in interactive mode.

Top

Parameters

Keyword	Description	Choices	Notes
HOSTNAME	Query name	*Character value*, ***NONE**	Optional, Positional 1
DMNNAMSVR	Domain name server	*Character value*, ***CFG**	Optional, Positional 2

Top

Query name (HOSTNAME)

Specifies the name that you want the Domain Name System (DNS) server to supply information about. You can use either a name or a numerical IP address. For example, a name could be ″mycomputer.mycompany.com″ or ″mycompany.com″. A numerical IP address could be ″64.236.1.107″. Use either the name to obtain the address, or use the address to obtain the name.

***NONE**
No host name is provided. If *NONE is specified or passed as the default for this parameter, an interactive NSLOOKUP session will be started.

Note: This value is only valid if the STRDNSQRY command is run in an interactive job.

host-name
Specify the name to use for the DNS server query.

host-internet-address
Specify the IPv4 address in dotted-decimal notation or a colon-delimited IPv6 address of a host to use for the DNS server query.

Top

Domain name server (DMNNAMSVR)

Specifies the name or the IP address of the DNS server that NSLOOKUP will use as its current server for the query session. You can specify any DNS server to which your TCP/IP network has access. Or, if you want to test the response of a DNS server prior to designating it for use by your system, specify that server.

NSLOOKUP retrieves information from DNS servers. It needs an active DNS server to send its queries to. If you do not specify a DNS server with DMNNAMSVR when you start the tool, it will attempt to set one of the following as its current DNS server for the session:

1. DNS server your system is configured to use (*CFG), or
2. The DNS server that is running on your local system.

If neither of these conditions exist, NSLOOKUP will not be able to retrieve any information until you specify a DNS server to query. DMNNAMSVR allows you to start the query session and set the DNS server of your choice as the current server for the session.

***CFG** Use the DNS server that is currently designated for use by this system. These server internet addresses can be seen by prompting the Change TCP/IP Domain (CHGTCPDMN) command and looking at the values shown for the INTNETADR parameter.

server-domain-name
Specify the name of a DNS server.

server-internet-address
Specify the IP address of a DNS server.

Top

Examples

Example 1: Simple DNS Lookup

```
STRDNSQRY   HOSTNAME('64.236.1.107')
            DMNNAMSVR(*CFG)
```

This command starts a DNS query using the host located at IP address 64.236.1.107 and the DNS server that is currently designated for use by this system.

The output from this type of query might look like this:

```
Server:         10.0.1.100
Address:        10.0.1.100#53
Non-authoritative answer:
107.1.236.64.in-addr.arpa       name = any-ext.ns.aol.com.
Authoritative answers can be found from:
1.236.64.in-addr.arpa   nameserver = dns-02.atdn.net.
1.236.64.in-addr.arpa   nameserver = dns-01.atdn.net.
dns-02.atdn.net internet address = 205.188.157.236
dns-01.atdn.net internet address = 152.163.159.236
```

Example 2: DNS Lookup of an Internet Address

```
NSLOOKUP HOSTNAME(aol.com)
         DMNNAMSVR('64.236.1.107')
```

This command starts a DNS query to the local machine to find the the internet address for **aol.com**.

The output from this type of query might look like this:

```
Server:         64.236.1.107
Address:        64.236.1.107#53
aol.com
        origin = dns-01.ns.aol.com
        mail addr = hostmaster.aol.net
        serial = 2006080400
        refresh = 1800
        retry = 300
        expire = 604800
        minimum = 600
```

Top

Error messages

***ESCAPE Messages**

DNS0013
Error processing command parameters.

DNS0065
Option 33 of i5/OS is required, but is not installed.

TCP7124
Program &1 in library &2 type *PGM ended abnormally.

Top

Start Disk Reorganization (STRDSKRGZ)

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

Parameters
Examples
Error messages

The Start Disk Reorganization (STRDSKRGZ) command allows the user to start the disk reorganization function for one or more auxiliary storage pools (ASPs). The user specifies a time limit that the function is to run for each ASP being reorganized. A message will be sent to the system history (QHST) log when the reorganization function is started for each ASP.

Unused space will be collected together within the ASP. This allows future large disk allocations to be done more efficiently.

Restriction: You must have *ALLOBJ special authority to use this command.

Top

Parameters

Keyword	Description	Choices	Notes
ASP	ASP number	Single values: *ALL Other values (up to 32 repetitions): 1-32	Optional, Positional 2
ASPDEV	ASP device	Values (up to 32 repetitions): *Name*, *ALLAVL	Optional
TIMLMT	Time limit	1-9999, *NOMAX	Required, Positional 1

Top

Auxiliary storage pool ID (ASP)

Specifies for which auxiliary storage pools the disk reorganization function is to be started.

***ALL**
Disk reorganization will be started for the system ASP (ASP number 1) and all basic ASPs (ASP numbers 2-32) defined to the system.

auxiliary-storage-pool-number
Specify the ASP for which disk reorganization is to be started. Valid ASP numbers are 1 to 32. Up to 32 ASP numbers may be specified.

Top

ASP device (ASPDEV)

Specifies the name of the auxiliary storage pool (ASP) device for which the disk reorganization is to be started. A value must be specified for the ASP parameter or the ASPDEV parameter.

***ALLAVL**
Disk reorganization will be started for all ASP devices that currently have a status of 'Available'.

auxiliary-storage-device-name

Specify the name of the independent ASP device for which disk reorganization is to be started. Up to 32 ASP device names may be specified.

Top

Time limit (TIMLMT)

Specifies the amount of time, in minutes, that the reorganization function is allowed to run. When the time limit is reached the function ends. The time limit specified is for each ASP being reorganized. For example, if ASP(*ALL) is specified and the machine has four ASP's configured and TIMLMT(60) is specified, four reorganization functions are started and each can run 60 minutes. If reorganization of any ASP has not completed after 60 minutes, it will be forced to end. This allows you to do disk reorganization incrementally.

***NOMAX**

There is no time limit for the reorganization function. For a large ASP that has many small unused disk areas, the reorganization function can take a long time to complete. If you start the reorganization function with *NOMAX and you want to force the function to end, you can use the End Disk Reorganization (ENDDSKRGZ) command.

time-limit

Specify the time limit that the reorganization function is allowed to run. Valid values range from 1 to 9999 minutes.

Top

Examples

Example 1: Starting Disk Reorganization for ASP 1

```
STRDSKRGZ   ASP(1)  TIMLMT(*NOMAX)
```

This command allows the user to start the disk reorganization function for ASP 1. The reorganization function will run until the ASP has been reorganized or it is ended through the ENDDSKRGZ command.

Example 2: Starting Disk Reorganization with a Time Limit

```
STRDSKRGZ   ASP(*ALL)  TIMLMT(60)
```

This command allows the user to start the disk reorganization function for each ASP on the system. Each reorganization function will have a time limit of sixty minutes. After sixty minutes, any reorganization functions which have not completed will be ended.

Example 3: Starting Disk Reorganization for an ASP Device

```
STRDSKRGZ   ASPDEV(MYASP1)  TIMLMT(*NOMAX)
```

This command allows the user to start the disk reorganization function for ASP device MYASP1. The reorganization function will run until the ASP has been reorganized or it is ended through the ENDDSKRGZ command.

Top

Error messages

***ESCAPE Messages**

CPF1888
Disk reorganization for ASP &1 already started.

CPF1890
*ALLOBJ authority required for requested operation.

CPF1891
Disk reorganization cannot be started on an ASP with only one unit.

Top

Start Disk Watcher (STRDW)

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

Parameters
Examples
Error messages

The Start Disk Watcher (STRDW) command starts the collection of disk performance data. It allows the user to obtain data including counts and durations of input/output (I/O) operations to disk units. In addition, it provides useful run-time data regarding which objects are being read from disk or written to disk, and which jobs, threads, or LIC tasks are causing the operations. The collected data is written to a set of database files. The file names all begin with the letters 'QAPYDW'.

Restrictions:

- To use this command, you must have service (*SERVICE) special authority, or be authorized to the Disk Watcher function of the Operating System through System i5 Navigator's Application Administration support. The Change Function Usage (CHGFCNUSG) command, with a function ID of QIBM_SERVICE_DISK_WATCHER, can also be used to change the list of users that are allowed to use this command.
- You must have execute (*EXECUTE) authority to the library specified in the **Library (LIB)** parameter.

Top

Parameters

Keyword	Description	Choices	Notes
DFN	Definition	*Name*, ***SELECT**	Optional
COL	Collection	*Name*, ***GEN**	Optional
LIB	Library	*Name*, **QPFRDATA**	Optional
TEXT	Text 'description'	*Character value*, ***DFN**, *BLANK	Optional
RPLDTA	Replace data	***NO**, *YES	Optional
COLITV	Collection interval	1-3600, ***DFN**, *DYN	Optional
ENDCOL	End collection	Values (up to 3 repetitions): *Element list*	Optional
	Element 1: Option	***NBRSEC**, *DASDMB, *NBRITV	
	Element 2: Value	*Integer*, **60**	
HDWF	Hardware file	***NO**, *YES	Optional

Top

Definition (DFN)

Specifies the Disk Watcher definition to use when starting a new Disk Watcher collection. The definition identifies the disk I/O data to be collected. A new Disk Watcher definition can be created using the **Add Disk Watcher Definition (ADDDWDFN)** command.

***SELECT**
The user will be prompted to select from a list of existing Disk Watcher definitions.

name Specify the name of the Disk Watcher definition to use for this collection.

Top

Collection (COL)

Specifies the name of the collection. The name of the collection is the name of the database file member to which the output data is to be written. If a member by this name does not exist in each Disk Watcher database file, one will be created with the specified name. If a member by this name already exists, you must specify *YES on the **Replace data (RPLDTA)** parameter in order to write over the data in the existing member.

***GEN** The collection name will be generated based on when the STRDW command was run. The format of the collection name will be **Qdddhhmmss** where **ddd** is the Julian day (001-366) and **hhmmss** is the hour (01-24), minute (00-59) and second (00-59) the disk I/O data collection was started.

name Specify the name of the collection.

Top

Library (LIB)

Specifies the library where the database files for the Disk Watcher data are to exist. Each file that is not found in the specified library is automatically created in that library.

QPFRDATA
The database files will be located in library QPFRDATA.

name Specify the name of the library where the database files will be located.

Top

Text ’description’ (TEXT)

Specifies the text to be used for each member across the set of Disk Watcher database files associated with the collection.

***DFN** Disk Watcher database file members will have the same text description as the Disk Watcher definition that was selected on the **Definition (DFN)** parameter.

***BLANK**
Disk Watcher database file members will have no text description.

character-value
Specify the text description for this set of Disk Watcher database file members. The description should be no more that 50 characters of text, enclosed in apostrophes.

Top

Replace data (RPLDTA)

Specifies whether an existing set of Disk Watcher database file members should be replaced with a new set of members with the same name.

***NO** The existing Disk Watcher database file members will not be replaced. A new Disk Watcher collection will not begin if the specified member already exists.

***YES** The existing Disk Watcher database file members will be replaced with data from the new collection.

Top

Collection interval (COLITV)

Specifies the interval between retrieval of disk I/O data.

***DFN** The collection interval specified in the Disk Watcher definition will be used for this collection.

***DYN** The time interval between samples is dynamic. This indicates that the interval seconds will be adjusted to the optimal value and may change during the collection. The optimal value is the largest interval time that can be used without missing data.

1-3600 Specify the number of seconds between intervals.

Top

End collection (ENDCOL)

Specifies the criteria which will end the collection. This parameter allows you to specify multiple conditions which will end the collection of data. If more than one ending criteria is specified, disk I/O data collection will end the first time any one of specified criteria has been met.

You can specify 3 values for this parameter.

Element 1: Option

***NBRSEC**
End the collection after a number of seconds has elapsed. Specify the time limit in element 2 of this parameter.

***DASDMB**
End the collection when a number of megabytes of data have been written to the Disk Watcher database files. Specify the megabyte limit in element 2 of this parameter.

***NBRITV**
End the collection when a number of collection intervals have occurred. Specify the number of intervals in element 2 of this parameter.

Element 2: Value

60 If the default of *NBRSEC is used for element 1 of this parameter, the collection will end after sixty seconds of disk I/O data is collected.

integer
Specify the number of seconds (for *NBRSEC) or the number of megabytes (for *DASDMB) or the number of intervals (for *NBRITV) to use as the ending criteria for disk I/O data collection.

Top

Hardware file (HDWF)

Specifies whether data will be written to the QAPYDWHDWR file for this collection. This file contains storage device resource information provided by the **Display Hardware Resources (DSPHDWRSC)** command.

***NO** Storage device resource information will not be written to the QAPYDWHDWR file for this collection.

***YES** Storage device resource information will be written to the QAPYDWHDWR file for this collection. The name of the QAPYDWHDWR file member will be the name specified on the **Collection (COL)** parameter.

Top

Examples

Example 1: Start a Disk Watcher Collection

```
STRDW   DFN(MYDFN)  COL(TEST)  LIB(MYLIB)
```

This command will start Disk Watcher using definition MYDFN to determine which data should be collected. Assuming the shipped default of ENDCOL(*NBRSEC 60) is used, data collection will end after 60 seconds. Collected data will be written to the Disk Watcher database files in library MYLIB in member TEST. The Disk Watcher database file names all begin with 'QAPYDW'.

Example 2: Start Disk Watcher Using a Collection Interval

```
STRDW   DFN(MYDFN)  COLITV(5)  ENDCOL((*NBRITV 200))
```

This command will start Disk Watcher to collect data using the definition called MYDFN. Data will be collected at an interval of 5 seconds, overriding the interval specified in the definition. Data collection will end after 200 intervals have been collected. Data will be written to the Disk Watcher database files in library QPFRDATA in a member name which will be generated by Disk Watcher based on the date and time the collection started.

Example 3: Start Disk Watcher, Prompting to Select a Definition

```
STRDW   DFN(*SELECT)  COL(TEST)  LIB(MYLIB)
        ENDCOL((*DASDMB 100))
```

This command will open a prompt panel listing all Disk Watcher definitions which currently exist on the system. The definition selected on this panel will be used to determine what data will be collected. Data collection will end after 100 megabytes of data have been written to the Disk Watcher database files in library MYLIB in member TEST.

Top

Error messages

***ESCAPE Messages**

CPF2401
Not authorized to library &1.

CPF9810
Library &1 not found.

CPFB511
Disk Watcher could not start due to reason code &1.

CPFB512
Start Disk Watcher did not complete successfully.

CPFB513
The user does not have the required authority.

CPFB514

Member already exists.

CPFB515

Not enough storage in machine pool.

CPFB51B

Option *SELECT is only valid if the command is being run interactively.

Top

Start Education (STREDU)

Where allowed to run:

- Interactive job (*INTERACT)
- Interactive program (*IPGM)
- Interactive REXX procedure (*IREXX)

Threadsafe: No

Parameters
Examples
Error messages

The Start Education (STREDU) command starts the online education session.

There are no parameters for this command.

Top

Parameters

None

Top

Examples

```
STREDU
```

This command shows the following menus:

- The **Start Education Administration** menu is shown for the Administrator.
- The **Select Course Option** menu is shown for the new student that was enrolled by the Administrator.
- The **Specify your Name** data entry screen is shown for the new student that was not enrolled.
- The **Select Course Option** menu is shown for the enrolled student.

Top

Error messages

None

Top

Start 3270 Display Emulation (STREML3270)

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

Parameters
Examples
Error messages

The Start 3270 Display Emulation (STREML3270) command starts a 3270 device emulation session for a display device to a binary synchronous communications (BSC) or Systems Network Architecture (SNA) host system. The user can type this command on the command line or from any display station that allows CL commands to be specified. This command can also be issued from a batch job by specifying the display device (DSPDEV) parameter.

The STREML3270 command can be in a CL program specified as the INLPGM for a user profile that is run when the user signs on the display device.

More information on device emulation is in the 3270 Device Emulation Support book, SC41-5408.

Top

Parameters

Keyword	Description	Choices	Notes
EMLCTL	Emulation controller, or	*Name*	Optional, Positional 1
EMLDEV	Emulation device, or	*Name*	Optional, Positional 2
EMLLOC	Emulation location	*Communications name*	Optional, Positional 3
DSPDEV	Display device, batch only	*Name*, ***CURRENT**	Optional
PAGEUP	Page Up (Roll Down) key	***PA2**, *PA1, *PA3, *NONE, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12, *F13, *F14, *F15, *F16, *F17, *F18, *F19, *F20, *F21, *F22, *F23, *F24, *CLEAR, *ERASEINP, *CSRSLT	Optional
PAGEDOWN	Page Down (Roll Up) key	***PA1**, *PA2, *PA3, *NONE, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12, *F13, *F14, *F15, *F16, *F17, *F18, *F19, *F20, *F21, *F22, *F23, *F24, *CLEAR, *ERASEINP, *CSRSLT	Optional
TESTREQ	Test Request key	***DFT**, *CLEAR, *ERASEINP	Optional
CSRSLT	Cursor Select key	***NONE**, *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12, *F13, *F14, *F15, *F16, *F17, *F18, *F19, *F20, *F21, *F22, *F23, *F24	Optional
IGCEMLPC	SNA DBCS 3270PC emulation	***NO**, *YES	Optional
EMLPRTDEV	Emulation printer device	*Name*, ***NONE**, *EMLCTL, *EMLLOC	Optional
INZWAIT	Timeout wait for host	1-32767, **120**, *NOMAX	Optional
NUMLCK	Numeric lock keyboard	***EMLDEV**, *NO, *YES	Optional
NULLS	Handle nulls	***BLANK**, *REMOVE	Optional
LOGON	Host signon/logon command	*Character value*, ***NONE**	Optional
WAITRSP	Wait response	***NO**, *YES	Optional

Keyword	Description	Choices	Notes
ENDCOND	End emulation conditions	Single values: ***NONE** Other values (up to 2 repetitions): *DACTLU, *UNBIND	Optional
ATNEMLMNU	Attention emulation menu	***YES**, *NO	Optional
FKEYPGM	Function key program	Single values: ***NONE** Other values: *Element list*	Optional
	Element 1: Program	*Qualified object name*	
	Qualifier 1: Program	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
	Element 2: Function keys	Single values: *ALLFKEYS Other values (up to 24 repetitions): *F1, *F2, *F3, *F4, *F5, *F6, *F7, *F8, *F9, *F10, *F11, *F12, *F13, *F14, *F15, *F16, *F17, *F18, *F19, *F20, *F21, *F22, *F23, *F24	
KBDTYPE	Keyboard language type	***DSPDEV**, *SYSVAL, *LCL, AGB, AGE, AGI, AGM, ALI, ALM, BGB, BGE, BLI, BLM, BRB, BRE, CAB, CAE, CAI, CAM, CLB, CLE, CSB, CSE, CYB, DMB, DME, DMI, DMM, ESB, FAB, FAE, FAI, FAM, FNB, FNE, FNI, FNM, FQB, FQI, GKB, GNB, GNE, HIB, HNB, HNE, ICB, ICE, ICI, ICM, INB, INI, IRB, ITB, ITE, ITI, ITM, JEB, JEI, JKB, JPB, JPE, JUB, KAB, KOB, LAE, LTB, LVB, MKB, MKE, NCB, NCE, NEB, NEE, NEI, NEM, NWB, NWE, NWI, NWM, PLB, PKE, PLE, PRB, PRE, PRI, PRM, RCB, RMB, RME, ROB, ROE, RUB, RUE, SFI, SFM, SGI, SGM, SKB, SKE, SPB, SPE, SPI, SPM, SQB, SQE, SSB, SSI, SSE, SSM, SWB, SWE, SWI, SWM, TAB, THB, THE, TKB, TKE, TRB, TRE, UAE, UKB, UKE, UKI, UKM, USB, USE, USI, USM, VNE, YGI, YGM, *TRNTBL	Optional
TRNTBLOUT	Outgoing translation table	Single values: ***KBDTYPE** Other values: *Qualified object name*	Optional
	Qualifier 1: Outgoing translation table	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
TRNTBLIN	Incoming translation table	Single values: ***KBDTYPE** Other values: *Qualified object name*	Optional
	Qualifier 1: Incoming translation table	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
EMLCFGE	Configuration entry	*Name*, QEMDFTCFGE, ***NONE**	Optional

Top

Emulation controller (EMLCTL)

Specifies the name of a binary synchronous communications (BSC) controller description or Systems Network Architecture (SNA) controller description that has attached 3270 emulation device descriptions. When this parameter is specified, the requesting display device is linked to an available 3270 emulation device on the emulation controller. At least one device attached to the controller must be available, and the requester of the command must be authorized to use the controller and device.

Either this parameter, the **Emulation device** prompt (EMLDEV parameter), or the **Emulation location** prompt (EMLLOC parameter) is required.

Top

Emulation device (EMLDEV)

Specifies the name of a binary synchronous communications (BSC) or any System Network Architecture (SNA) device emulation (3278) that is linked to the requesting display device to emulate a 3270 display device. The requester must be authorized to this device, and the device must be available.

Either this parameter, the **Emulation controller (EMLCTL)** parameter, or the **Emulation location (EMLLOC)** parameter is required.

Top

Emulation location (EMLLOC)

Specifies the remote location name that describes the location of the 3270 display emulation devices. This name is defined during device description configuration, and it refers to the remote location where communication takes place. When this parameter is specified, the requesting display device is linked to an available 3270 emulation device referred to by the remote location. At least one of the emulation devices referred to by the remote location must be available, and the requester of the command must be authorized to use the device. A remote location can refer to as many as 1,016 emulation display devices.

Either this parameter, the **Emulation controller** prompt (EMLCTL parameter), or the **Emulation device** prompt (EMLDEV parameter) is required.

Top

Display device, batch only (DSPDEV)

Specifies the name of the display device used for display emulation when the command is in a batch job. The 3270 support tries to acquire the display device by this name; if the display device is acquired, the 3270 device emulation is active on that display device.

***CURRENT**
The current display device name is used for device emulation. This parameter is used when the command is in an interactive job.

display-device-name
Specify the display device name used for device emulation. This parameter is used when the command is in a batch job.

Top

Page Up (Roll Down) key (PAGEUP)

Specifies a 3270 function for the Page Up (Roll Down) key on the 5250 type keyboard when 3270 device emulation is active. This assignment is in effect when the number of input fields is not larger than the maximum number of input fields.

***PA-key**
Specify the 3270 PA key assigned to the Page Up (Roll Down) key. The default is the *PA2 key.

***NONE**
No function is assigned to the Page Up (Roll Down) key. When there are fewer input fields on the display than allowed by the 5250 display device, this key has no function.

***F-key** Specify the 3270 F key assigned to the Page Up (Roll Down) key.

***CLEAR**
The 3270 CLEAR key is assigned to the Page Up (Roll Down) key.

***ERASEINP**
The 3270 ERASE INPUT key is assigned to the Page Up (Roll Down) key.

***CSRSLT**
The 3270 CURSOR SELECT key is assigned to the Page Up (Roll Down) key and does not allow the real Cursor Select key to be used.

Top

Page Down (Roll Up) key (PAGEDOWN)

Specifies an added function for the Page Down (Roll Up) key on the 5250 type keyboard when 3270 device emulation is active. This assignment is in effect when the number of input fields is not larger than the maximum number of input fields.

***PA-key**
Specify the 3270 PA key assigned to the Page Down (Roll Up) key. The default is the *PA1 key.

***NONE**
No function is assigned to the Page Down (Roll Up) key. When there are fewer input fields on the display than allowed by the 5250 display device, the key has no function.

***F-key** Specify the 3270 F key assigned to the Page Down (Roll Up) key.

***CLEAR**
The 3270 CLEAR key is assigned to the Page Down (Roll Up) key.

***ERASEINP**
The 3270 ERASE INPUT key is assigned to the Page Down (Roll Up) key.

***CSRSLT**
The 3270 CURSOR SELECT key is assigned to the Page Down (Roll Up) key and does not allow the real Cursor Select key to be used.

Top

Test Request key (TESTREQ)

Specifies an added function for the Test Request key on the 5250 keyboard when 3270 device emulation is active.

***DFT** Normal function is assigned to the Test Request key. This is the system default. The normal function depends on whether the 3270 emulation display device uses binary synchronous communications (BSC) or Systems Network Architecture (SNA) protocol. BSC defaults to a 3270 Test Request function, while SNA defaults to a 3270 System Request function.

***CLEAR**
The 3270 CLEAR key is assigned to the Test Request key.

***ERASEINP**
The 3270 ERASE INPUT key is assigned to the Test Request key.

Top

Cursor Select key (CSRSLT)

Specifies one of several physical function keys to be used as the Cursor Select key. When 3270 emulation is active, the specified key can be used to select or reject selectable fields.

***NONE**
A physical function key is not assigned to emulate the cursor select key. The real Cursor Select key is used.

***F-key** Specify the function key assigned to emulate the Cursor Select key. The use of the real Cursor Select key is not allowed.

Top

SNA DBCS 3270PC emulation (IGCEMLPC)

Specifies whether System Network Architecture (SNA) double-byte character set (DBCS) 3270PC emulation or 3270 device emulation is used. This parameter is valid only when using a S/55 Personal Computer.

***NO** SNA DBCS 3270PC emulation is not used.

***YES** SNA DBCS 3270PC emulation is used.

Top

Emulation printer device (EMLPRTDEV)

Specifies the emulation printer device that is used for SNA DBCS 3270PC emulation. The printer emulation device is selected after the device emulation is selected. This parameter is valid only when *YES is specified on the **SNA DBCS 3270PC emulation** prompt (IGCEMLPC parameter).

***NONE**
SNA DBCS 3270PC printer emulation with device emulation is not used. No printer emulation device is selected.

emulation-device-description-name
Specify the printer emulation device with the selected display emulation device for SNA DBCS 3270PC emulation.

***EMLCTL**
The first available printer emulation device on the specified controller on the **Emulation controller** prompt (EMLCTL parameter) is used.

***EMLLOC**
The first available printer emulation device from the specified location on the **Emulation location** prompt (EMLLOC parameter) is used.

Top

Timeout wait for host (INZWAIT)

Specifies the initial amount of time (in seconds) that 3270 emulation waits for the first display data from the host system. If the host system does not send the first display in this time, the emulation session is ended, and a message is returned to the requester.

120 3270 emulation waits 120 seconds for the first display from the host system.

***NOMAX**

There is no limit on the amount of time 3270 emulation waits for the first display from the host system. This value can be used when the user is not sure when the host system is active to this session. The request can be ended by using the system request and ending request functions.

number-of-seconds

Specify the length of time (in seconds) that the 3270 emulation waits for the first display from the host system. Valid values range from 1 through 32767 seconds.

Top

Numeric lock keyboard (NUMLCK)

Specifies whether numeric input fields will only allow numeric data on a 5250 keyboard.

***EMLDEV**

Numeric lock is specified in the EMLNUMLCK field of the emulation device description. You can use the DSPDEVD command to display the current EMLNUMLCK value for the emulation device. The value can be changed using the BSC commands CRTDEVBSC or CHGDEVBSC or the SNA commands CRTDEVHOST or CHGDEVHOST.

***NO** 3270 emulation will allow any data to be typed in the numeric input fields.

***YES** 3270 emulation will only allow numeric data to be typed in the numeric input fields. Numeric data that can be typed include the characters 0 through 9, and symbols ″ + - , . ″ and the blank symbol, which is the character *b* with a slash on the stem.

Top

Handle nulls (NULLS)

Specifies how beginning and embedded nulls within the 3270 data stream sent from a 5250 display station are handled. Beginning nulls are those that occur before a character that is not null. Embedded nulls are those that occur between characters that are not null.

***BLANK**

Beginning and embedded nulls are changed to blanks within the 3270 data stream.

***REMOVE**

Beginning and embedded nulls are removed from the 3270 data stream.

Top

Host signon/logon command (LOGON)

Specifies the sign-on text that is sent to the host system after SNA 3270 emulation is started. This text can be used to sign on to a specific host application.

This parameter is not allowed if specified for BSC 3270 emulation, SNA 3270 display station pass-through, or SNA DBCS 3270PC emulation.

***NONE**

No text is sent to the host system after 3270 emulation is started.

host-logon-command

Specify text that is sent to the host system after 3270 emulation is started. The text must be enclosed in apostrophes if it contains blanks or other special characters. All apostrophes within the text must be represented by two apostrophes. A maximum of 256 characters can be specified.

Top

Wait response (WAITRSP)

Specifies whether the 3270 emulation device waits until the data received is shown on the workstation display to send a positive response to the host system. The response time recorded by the System i5 may be longer than the time recorded by the host when the emulation device does not wait.

***NO** The emulation device does not wait to send a positive response. It sends the response as soon as the data is received to the workstation display.

***YES** The emulation device waits until the data received is shown on the workstation display to send a positive response.

Top

End emulation conditions (ENDCOND)

Specifies additional ways in which the SNA 3270 device emulation session can end.

This parameter is not allowed if specified for BSC 3270 emulation, SNA 3270 display station pass-through, or SNA DBCS 3270PC emulation.

The possible values are:

***NONE**
No additional ways to end 3270 device emulation are requested.

***DACTLU**
The 3270 emulation session will end if it receives an SNA DACTLU from the host system. Please consider the following before selecting this end condition:

- There are certain host system applications that issue a DACTLU before starting, such as Time Sharing Option (TSO), which will end the 3270 emulation session before the desired application is accessed. This end condition should be avoided when trying to access these applications.

***UNBIND**
The 3270 emulation session will end if it receives an SNA UNBIND from the host system. Consider the following items before selecting this end condition:

- This end condition should be used only when you need to access one host application for the duration of the session. An UNBIND will occur while switching from one application to the next, and the 3270 session will end before accessing the second application.
- This end condition should only be used when the communication path to the host system is a simple one. A simple communication path is one that only involves accessing the System i5 where the Start 3270 Display Emulation (STREML3270) command is run, and accessing the host system that contains the desired application. Intermediate systems can exist along this simple path as long as they are not accessed. If intermediate systems are accessed, an UNBIND will occur while switching from one system to the next, and the 3270 display emulation session will end before accessing the desired application.
- There are certain host system applications that issue an UNBIND before starting, such as Time Sharing Option (TSO), which will end the 3270 display emulation session before the desired application is accessed. This end condition should be avoided when trying to access these applications.

Top

Attention emulation menu (ATNEMLMNU)

Specify whether you want the Select 3270 Emulation Option for SNA menu or the Select 3270 Emulation Option for BSC menu to be displayed when the Attention key is pressed.

This parameter is not allowed if specified for either SNA 3270 display station pass-through or SNA DBCS 3270PC emulation.

The possible values are:

***YES** The Select 3270 Emulation Option for SNA or BSC menu is displayed when you press the Attention key.

***NO** The Select 3270 Emulation Option for SNA or BSC menu is not displayed when you press the Attention key. The attention program (if any) currently active in the job will get control when the Attention key is pressed. You can still display the Select 3270 Emulation Option for SNA or BSC menu by pressing the System Request key sequence and then selecting the **Display 3270 emulation options menu** option from the System Request menu.

Top

Function key program (FKEYPGM)

Specifies a user-exit program and one or more function keys that call the program. When a specified function key is pressed during the 3270 display emulation session and is sent to the host system, the user-exit program is called. When the user-exit program ends, control is returned to the 3270 display emulation session at the point where the function key was pressed.

This parameter is not valid if specified for either BSC 3270 display emulation, SNA 3270 display station pass-through, or SNA DBCS 3270PC emulation.

The user-exit program is called only if the function key is successfully sent to the host system. If the function key fails to be received, an error reset message appears at the bottom of the display suggesting you try again.

The user-exit program must be coded to allow for input parameters. The following parameters are passed to the program in the specified order:

1. The function key identifier (10 characters). The identifier of the function key that was pressed. If function key 1 is pressed, the parameter value is *F1. If function key 2 is pressed, the parameter value is *F2, and so on, up to function key 24. The value is left-justified within the parameter.
2. The display name (10 characters). The name of the display on which the 3270 display emulation running. The value is left-justified within the parameter.
3. The cursor location (6 characters). The screen location of the cursor at the time the function key was pressed. The first three characters are the row position of the cursor location. The second three characters are the column position of the cursor location. For example, if the cursor location is row 24, column 1 when the function key is pressed, the value of the parameter is 024001. The row and column can be extracted from the variable using substring logic.

The possible **program name** values are:

***NONE**
A user-exit program is not associated with any function key.

program-name
Specify the name and library of the user-exit program that is called when one of the specified function keys is pressed. The program cannot be a system program.

The possible library values are:

***LIBL** All libraries in the user and system portions of the job's library list 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 to be searched.

The possible **function key** values are:

***ALLFKEYS**
All function keys call the specified user-exit program.

function-key
Specify a function key to call the user-exit program. A maximum of 24 values can be specified on this parameter.

Top

Keyboard language type (KBDTYPE)

Specifies the 3-character keyboard language identifier which represents a specific full character identifier (CHRID - comprised of a character set and code page) that is used on the display station. To determine the full CHRID from the keyboard language identifier, see the CHRID Values table in the Create Device Display (CRTDEVDSP) command in the CL Reference.

This parameter does not apply when running SNA 3270 display station pass-through.

***DSPDEV**
If a local display device is specified for the STREML3270 display device (DSPDEV) parameter, then use the specified display's device description current KBDTYPE value. If a remote display device is specified for the STREML3270 DSPDEV parameter, then use the current QKBDTYPE system value.

***SYSVAL**
Use the current QKBDTYPE system value. This value is valid for both local and remote displays.

***LCL** The display device that requested 3270 device emulation is a local display device. The keyboard type is determined from the display device description.

***TRNTBL**
Allows user-defined translation tables to be used. The character translation is defined in the translation tables specified by the **Outgoing translation table** prompt (TRNTBLOUT parameter) and the **Incoming translation table** prompt (TRNTBLIN parameter).

If a local display device is specified for the STREML3270 Display Device (DSPDEV) parameter, then use the specified display's device description current KBDTYPE value to determine the CHRID to be used. If a remote display device is specified for the STREML3270 DSPDEV parameter, then use the current QKBDTYPE system value.

keyboard-language-identifier
Specify the keyboard language identifier to be used.

Top

Outgoing translation table (TRNTBLOUT)

Specifies the outgoing translation table that is used to translate characters sent from the host system to 3270 Emulation. If *TRNTBL is specified on the **Keyboard language type** prompt (KBDTYPE parameter), the **Incoming translation table** prompt (TRNTBLIN parameter), must also be specified.

***KBDTYPE**
Translation is done using the language specified on the **Keyboard language type** prompt (KBDTYPE parameter).

table-name
Specify the name and library of the table that is used for outgoing translation.

The possible library values are:

***LIBL** All libraries in the user and system portions of the job's library list are searched until the first match is found.

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

name Specify the name of the library to be searched.

Top

Incoming translation table (TRNTBLIN)

Specifies the incoming translation table that is used to translate characters sent from 3270 Emulation to the host system. If *TRNTBL is specified on the **Keyboard language type** prompt (KBDTYPE parameter), the **Outgoing translation table** prompt (TRNTBLOUT parameter), must also be specified.

***KBDTYPE**
Translation is done using the language specified on the **Keyboard language type** prompt (KBDTYPE parameter).

table-name
Specify the name and library of the table used for incoming translation.

The possible library values are:

***LIBL** All libraries in the user and system portions of the job's library list are searched until the first match is found.

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

name Specify the name of the library to be searched.

Top

Configuration entry (EMLCFGE)

Specifies whether a configuration entry is used for this session. Configuration entries indicate 3270 emulation configuration options. Configuration entries are created with the Add Emulation Configuration Entry (ADDEMLCFGE) command.

The possible values are:

***NONE**
No configuration entry is named and the configuration entry defaults are used.

QEMDFTCFGE
The default configuration entry QEMDFTDFGE is used. This entry is shipped with configuration entry defaults, and can be updated with the Change Emulation Configuration Entry (CHGEMLCFGE) command.

configuration-entry-name
Specify the name of the configuration entry to be used. If the configuration entry named does not exist in the configuration file, the configuration entry defaults are used.

Top

Examples

```
STREML3270   EMLCTL(TSOHOST)  PAGEUP(*F7)  PAGEDOWN(*F8)
```

This command places the display device into an emulation session that uses the first available device on the controller description TSOHOST for which the user has authority. When there are fewer input fields on the display than the maximum allowed by the 5250 display device and the Page Up key is pressed, an F7 key value is sent to the host system. When the Page Down key is pressed, an F8 key value is sent to the host system.

Top

Error messages

***ESCAPE Messages**

CPF2619
Table &1 not found.

CPF269A
Library parameter is not set to ″QSYS ″ on call.

CPF269B
T.61 conversion table not found.

CPF85EA
Screen address received from host is larger than screen size.

CPF85EB
3270 device emulation session ended.

CPF85EC
Specifying text on the LOGON parameter is not supported.

CPF85ED
Values other than ENDCOND(*NONE) are not supported.

CPF85E2
3270 display emulation is already active at this job.

CPF85E4
Not authorized to translation table &1 in library &2.

CPF85E5
3270 emulation device &1 is reserved for device &2.

CPF85E6
Translation table &1 in library &2 was not found.

CPF8503
Emulation controller &1 not found.

CPF8504
Controller &1 does not support 3270 emulation.

CPF8505
Emulation device &2 not found.

CPF8506
Emulation location &1 not found.

CPF8507
Display emulation cannot open required file.

CPF8508
Host system did not respond.

CPF851A
Maximum number of shift in and shift out characters exceeded.

CPF8510
Internal error occurred on device &1.

CPF8511
Emulation ended by errors on device &2.

CPF8512
Emulation ended because device &2 was held.

CPF8513
Emulation ended by errors on device &2.

CPF8514
Error recovery stopped on device &1.

CPF8515
3270 emulation session ended by host.

CPF8516
No match between host and device &2.

CPF8517
Received more than maximum number of fields allowed.

CPF8518
Emulation ended because of internal failure in system.

CPF8519
Function check in 3270 emulation.

CPF852A
Values other than FKEYPGM(*NONE) are not supported.

CPF852B
Program &1 not found.

CPF852C
Not authorized to program &1.

CPF8521
Not authorized to controller &1.

CPF8522
Not authorized to emulation device &2.

CPF8523
*NO on the ATNEMLMNU parameter is not supported.

CPF8524
Emulation cannot open its required display file.

CPF8525
KBDTYPE(*LCL) not allowed for remote display devices.

CPF8526
No 3270 display emulation devices available.

CPF8527
Emulation device &2 not available.

CPF8528
Device &2 is not a display emulation device.

CPF8530
Not authorized to use any display emulation device.

CPF8533
Display device not specified in a batch job.

CPF8534
Display device &1 is not available.

CPF8535
Display device &1 not found.

CPF8536
Not authorized to display device &1.

CPF8539
&1 keyboard type not supported by 3270 emulation.

CPF8546
No 3270 display emulation devices available.

CPF8547
No 3270 printer emulation devices available.

CPF8550
Emulation ended due to time-out internal failure.

CPF8551
Emulation ended with error code &1.

CPF8552
Emulation ended because of return code.

CPF8553
BSC controller or device not allowed without translation.

CPF8565
Emulation device &1 not found.

CPF8568
Device &1 not printer emulation device.

CPF8569
Not authorized to use any printer emulation device.

CPF8571

No 3270 printer emulation devices available.

CPF8572

Emulation device &1 not available.

Top

Start EPM Environment (STREPMENV)

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

Parameters
Examples
Error messages

The Start EPM Environment (STREPMENV) command is used with extended program model (EPM) languages to create a user-controlled environment. The C/400*, FORTRAN/400*, and Pascal languages are part of the extended program model.

You can use this command to create a run-time environment for an EPM language entry point that you are calling from another language program. See the Extended Program Model User's Guide and Reference for more detailed information on the EPM and this command.

Error messages for STREPMENV

None

Top

Parameters

Keyword	Description	Choices	Notes
EPMENV	Environment Name	*Character value*	Required, Positional 1
ROOTPGM	Environment Program	*Qualified object name*	Required, Positional 2
	Qualifier 1: Environment Program	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	

Top

Environment Name (EPMENV)

Specifies the name of the user-controlled environment that is to be created. The environment name must be unique.

environment-name
Enter a name for the environment.

Top

Environment Program (ROOTPGM)

Specifies the name of the program and library that contains the environment definition information that is necessary in order to create the EPM run-time environment.

program-name
Enter the name of the EPM language program object that contains the environment definition information.

The possible values for library are:

***LIBL** The library list is searched to locate the specified program object.

***CURLIB**
The current library is searched to locate the specified program object.

library-name
Enter the name of the library that contains the specified program object.

Top

Examples

None

Top

Error messages

None

Top

Start Font Management Aid (STRFMA)

Where allowed to run: Interactive environments (*INTERACT *IPGM *IREXX *EXEC)
Threadsafe: No

Parameters
Examples
Error messages

The Start Font Management Aid (STRFMA) command displays the Font Management Aid (FMA) menu. From this menu, you can ″Work With″ user defined characters (24 X 24 dot matrix) in the DBCS font table with a workstation font file ($SYS1Z24.FNT). ″Work With″ means to copy user defined characters in the workstation font file to DBCS font table, or copy user defined characters in DBCS font table to workstation font file. FMA is also used to get a copy of the workstation user-font/dictionary file from the other workstation.

There are no parameters for this command.

Error messages for STRFMA

None

Top

Parameters

None

Top

Examples

None

Top

Error messages

None

Top

Start HOST Query (STRHOSTQRY)

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

Parameters
Examples
Error messages

The Start HOST Query (STRHOSTQRY) command, or its alias HOST, is a simple utility for performing Domain Name System (DNS) lookups. It is normally used to convert names to IP addresses and vice versa.

Restrictions:

- You must have execute (*X) authority to the directories in the path of the output file.
- You must have write (*W) authority to the output file if it already exists.
- You must have read, write and execute (*RWX) authority to the output file's parent directory if the output file does not already exist.

Top

Parameters

Keyword	Description	Choices	Notes
HOSTNAME	Query name	*Character value*	Required, Positional 1
TYPE	Query type	***A**, *AAAA, *ANY, *AXFR, *CNAME, *MX, *NS, *PTR, *SOA, *SRV, *TXT	Optional, Positional 2
CLASS	Query class	***IN**, *CH, *HS, *ANY	Optional
DMNNAMSVR	Domain name server	*Character value*, ***CFG**	Optional
SOA	Display SOA records	***NO**, *YES	Optional
AXFR	List all hosts	***NO**, *YES	Optional
IP6INT	Use IP6.INT domain	***NO**, *YES	Optional
SETRDFLAG	Recursion desired	***YES**, *NO	Optional
PROTOCOL	Network protocol	***UDP**, *TCP	Optional
DEBUG	Show debug information	***NO**, *YES	Optional
IPVSN	IP Version	***ALL**, *IPV4ONLY, *IPV6ONLY	Optional
NBRDOTS	Number of dots	0-10, **1**	Optional
TIMEOUT	Query timeout	1-100, **5**	Optional
UDPNBRRTY	UDP retry	0-100, **2**	Optional
TOSTMF	Output file	*Path name*, ***STDOUT**	Optional

Top

Query name (HOSTNAME)

Specifies the name that you want the Domain Name System (DNS) server to look up. It can also be a dotted-decimal IPv4 address or a colon-delimited IPv6 address, in which case HOST will by default perform a reverse lookup for that address.

This is a required parameter.

domain-name
Specify a valid domain name.

internet-address
Specify a valid IPv4 or IPv6 address.

Top

Query type (TYPE)

Specifies the type of the query. The default type is *A. The list of supported query types changes with time, and not all servers support all the types that exist. This list provides some common query types for convenience and is not a complete list. If you do not see a query type you want in this list, you can still type in a character string that represents that query type. If the type is unknown by this BIND version, the query will default to an *A query with any corresponding results.

<u>*A</u> IPv4 Address record. This is the character string ’a’.

***AAAA**
IPv6 address record. This is the character string ’aaaa’.

***ANY** Any resource record. This is the character string ’any’.

***AXFR**
Zone transfer. This is the character string ’axfr’.

***CNAME**
Canonical name record. Returns a list of aliases for the true (canonical) host name, if any exist. This is the character string ’cname’.

***MX** Mail exchange record. This is the character string ’mx’.

***NS** Name server (DNS server) information for the zone. This is the character string ’ns’.

***PTR** Pointer record. Returns a name for an IP address. This is the character string ’ptr’.

***SOA** Start of authority record. This is the character string ’soa’.

***SRV** Services location selection. This is the character string ’srv’.

***TXT** Text record. This is the character string ’txt’.

Top

Query class (CLASS)

Specifies the protocol group of the information.

<u>*IN</u> The Internet class.

***CH** The CHAOS class. The server provides some helpful diagnostic information through a number of built-in zones under the CHAOS class.

version
If your BIND server named.conf has the ’version’ option configured, then you can query for it using the CHAOS class. This is the version the server should report via a query of the name ’version.bind’ with type TXT, class CHAOS. The default is the real version number of this server.

```
HOST   HOSTNAME('version.bind')  TYPE(*TXT)  CLASS(*CH)
```

hostname
If your BIND server named.conf has the ’hostname’ option configured, then you can query for it using the CHAOS class. This is the hostname the server should report via a

query of the name ’hostname.bind’ with type TXT, class CHAOS. This defaults to the hostname of the machine hosting the name server. The primary purpose of such queries is to identify which of a group of anycast servers is actually answering your queries.

```
STRHOSTQRY   NAME('hostname.bind')  TYPE(*TXT)  CLASS(*CH)
```

server-id
If your BIND server named.conf has the ’server-id’ option configured, then you can query for it using the CHAOS class. This is the ID of the server should report via a query of the name ’ID.SERVER’ with type TXT, class CHAOS. The primary purpose of such queries is to identify which of a group of anycast servers is actually answering your queries. The default server-id is none.

```
HOST   HOSTNAME('ID.SERVER')  TYPE(*TXT)  CLASS(*CH)
```

***HS** The MIT Athena Hesiod class. Hesiod, developed by MIT Project Athena, is an information service built upon BIND. Its intent is to furnish information about users, groups, network-accessible file systems, printcaps, and mail service throughout an installation. In other words, it holds arbitrary data stored as text strings. Queries that use the HS class retrieve data stored as TXT Resource Records. You cannot specify root servers for the Hesiod class (generally the ones at MIT) in your list of root name servers.

***ANY** Query any class. This is a wildcard query.

Top

Domain name server (DMNNAMSVR)

Specifies the name or the IP address of the DNS server that HOST will use as its current server for the query session. You can specify any DNS server to which your TCP/IP network has access.

HOST retrieves information from DNS servers. It needs an active DNS server to send its queries to. If you do not specify a DNS server with DMNNAMSVR when you start the tool, it will attempt to set one of the following as its default DNS server for the session:

1. DNS server your system is configured to use (*CFG), or
2. The DNS server that is running on your local system.

***CFG** Use the DNS server that is currently designated for use by this system. These server internet addresses can be seen by prompting the Change TCP/IP Domain (CHGTCPDMN) command and looking at the values shown for the INTNETADR parameter.

server-domain-name
Specify the name of a DNS server.

server-internet-address
Specify the IP address of a DNS server.

Top

Display SOA records (SOA)

Specifies whether or not to print the SOA records for zone name from all the listed authoritative name servers for that zone. The list of name servers is defined by the NS records that are found for the zone.

***NO** Do not print SOA records.

***YES** Attempt to print SOA records.

Top

List all hosts (AXFR)

Specifies whether or not to make HOST perform a zone transfer for zone specified in the HOSTNAME parameter. A zone transfer is when all domain information in the zone is returned. HOST will display the NS, PTR and address records (A/AAAA).

*NO Do not perform a zone tranfer.

*YES Perform a zone transfer.

Top

Use IP6.INT domain (IP6INT)

Specifies whether or not to qualify the reverse lookup to be in the ip6.int zone and not the ip6.arpa zone.

*NO The normal ip6.arpa zone reverse lookup will be performed.

*YES An IPv6 address reverse lookup in the ip6.int zone will be performed. This zone is deprecated, but may still be required to query IPv6 backbone prefixes.

Top

Recursion desired (SETRDFLAG)

Specifies whether or not to set the Recursion Desired (RD) flag in the query. This should mean that the name server receiving the query will not attempt to resolve name. This enables HOST to mimic the behaviour of a name server by making non-recursive queries and expecting to receive answers to those queries that are usually referrals to other name servers.

*YES Set the RD flag.

*NO Do not set the RD flag.

Top

Network protocol (PROTOCOL)

Specified whether to use TCP or UDP when sending requests to the server.

*UDP Use UDP to send the query. However, TCP will be automatically selected for queries that require it, such as zone transfer (AXFR) requests.

*TCP Use TCP to send the query.

Top

Show debug information (DEBUG)

Specifies whether or not to turn debugging mode on. More information is displayed about the packet sent to the server and the resulting answer when debugging mode is on.

*NO Turn off debugging messages.

*YES Turn on debugging messages.

Top

IP Version (IPVSN)

Specifies whether to limit the query to IPv4 or IPv6 networks.

***ALL** Do not limit queries to IPv4 or IPv6.

***IPV4ONLY**
Only send queries out IPv4 network interfaces.

***IPV6ONLY**
Only send queries out IPv6 network interfaces.

Top

Number of dots (NBRDOTS)

Specifies the number of dots (period characters) that have to appear in a domain name for it to be considered absolute. Names with fewer dots are interpreted as relative names and will be searched for in the domains listed in the DMNSCHLIST parameter.

1 If the domain name contains one dot it is considered an absolute name.

0-10 Specify the number of dots that must be in a domain name for the domain name to be considered to be an absolute name.

Top

Query timeout (TIMEOUT)

Specifies the timeout interval, in seconds, to wait for a response.

5 A timeout value of 5 seconds is used.

1-100 Specify a valid timeout value in seconds.

Top

UDP retry (UDPNBRRTY)

Specifies the number of times to retry UDP queries to the current DNS server before attempting TCP queries.

2 Two UDP retries will be used. This is in addition to the initial query.

0-100 Specify the number of UDP retries. You normally only need to retry a query 3 to 5 times.

Top

Output file (TOSTMF)

Specifies the name of a stream file where all command output is written.

***STDOUT**
All command output goes to the standard output device (normally the display).

path-name
Specify the path for a stream file where output should be written.

Top

Examples

Example 1: Looking Up Internet Address for Domain Name

```
STRHOSTQRY  HOSTNAME(ibm.com)
```

This command attempts to find information about the domain 'aol.com'. This includes A and MX records, althought the output is formatted in a different style than other query tools (e.g. DIG). Sample output from this query might look like this:

```
Using domain server:
Name: 9.5.176.200
Address: 9.5.176.200#53
Aliases:
aol.com has address 64.12.50.151
aol.com has address 205.188.142.182
aol.com mail is handled by 15 mailin-01.mx.aol.com.
aol.com mail is handled by 15 mailin-02.mx.aol.com.
aol.com mail is handled by 15 mailin-03.mx.aol.com.
aol.com mail is handled by 15 mailin-04.mx.aol.com.
```

Example 2: Lookup with a Zone Transfer

```
HOST   HOSTNAME(i5os.ibm.com)  AXFR(*YES)
```

This command attempts to do a zone transfer. Sample output from this query might look like this:

```
Using domain server:
Name: 10.0.1.100
Address: 10.0.1.100#53
Aliases:
i5os.ibm.com name server MYDNS1.IBM.COM.
i5os.ibm.com name server MYDNS2.IBM.COM.
i5os.ibm.com has address 10.0.1.100
i5os.ibm.com has address 10.0.2.200
box1.i5os.ibm.com has address 10.0.2.201
box2.i5os.ibm.com has address 10.0.2.202
```

This command attempts to do the same zone transfer, but fails because the user is not authorized to do transfers. Sample output from this query might look like this:

```
Using domain server:
Name: 10.0.1.100
Address: 10.0.1.100#53
Aliases:
Host i5os.ibm.com not found: 9(NOTAUTH)
; Transfer failed.
```

Top

Error messages

*ESCAPE Messages

DNS0013
Error processing command parameters.

DNS0065

Option 33 of i5/OS is required, but is not installed.

TCP7124

Program &1 in library &2 type *PGM ended abnormally.

Top

Start Host Server (STRHOSTSVR)

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

Parameters
Examples
Error messages

The Start Host Server (STRHOSTSVR) command is used to start the optimized host server daemons and the server mapper daemon.

There is one server daemon for each of the host server types. In addition, there is one server mapper daemon for all host servers which provides support for client applications to obtain a particular host server daemon's port number. This port number is then used by the client application to connect to the host server's daemon. The daemon accepts the incoming connection request and routes it to the server job for further processing.

The daemons are batch jobs submitted to either the QSYSWRK or QSERVER subsystem, depending on the value or values specified for the SERVER keyword. All daemon jobs are submitted to the QSYSWRK subsystem with the exception of the *DATABASE and *FILE server daemons which are submitted to the QSERVER subsystem.

In order for the server daemons and the server mapper daemon to start successfully, the QSYSWRK subsystem and, for *DATABASE and *FILE server, the QSERVER subsystem must be active. If the required subsystem is not active, then the submission of the daemon job will fail. Additionally, the QUSRWRK subsystem or the user-defined subsystem must be active in order to start the associated server jobs. All associated server jobs can run in the QUSRWRK subsystem or a user-defined subsystem, except for the following:

- Server jobs QPWFSERVSO and QPWFSERVSS - these jobs run in the QSERVER subsystem or a user-defined subsystem
- Server job QPWFSERVS2 - this job runs in the QSERVER subsystem
- Server job QIWVPPJT - this job runs in the QSYSWRK subsystem
- Server job QTFPJTCP - this job runs in the QSERVER subsystem.

There are no server jobs associated with the server mapper daemon.

The server daemons must be active in order to allow client applications to establish a connection with the host server using sockets communication support. Once started, the server daemons and the server mapper daemon remain active until they are ended explicitly using the End Host Server (ENDHOSTSVR) command or an error occurs.

Restrictions:

- This command is used only for enabling client applications to communicate with any of the host servers using sockets communication support. This command does not start any of the APPC host servers; these are started as a result of a program start request.
- Only one server daemon can be active for a specific server type. Requests to start a server daemon that is already active will result in an informational message to the user issuing this command.

Error messages for STRHOSTSVR

***ESCAPE Messages**

PWS300D

Unable to start host server daemon jobs.

PWS3006
Errors occurred starting server daemon jobs.

Top

Parameters

Keyword	Description	Choices	Notes
SERVER	Server type	Single values: *ALL Other values (up to 8 repetitions): *CENTRAL, *DATABASE, *DTAQ, *FILE, *NETPRT, *RMTCMD, *SIGNON, *SVRMAP	Required, Positional 1
RQDPCL	Required protocol	*ANY, *NONE, *TCP	Optional

Top

Server type (SERVER)

Specifies the host server daemons to be started by this command.

The possible values are:

***ALL** All of the server daemons and the server mapper daemon are started.

***CENTRAL**
The central server daemon is started in the QSYSWRK subsystem. The daemon job is named QZSCSRVSD. The associated server job is named QZSCSRVS.

***DATABASE**
The database server daemon is started in the QSERVER subsystem. The daemon job is named QZDASRVSD. The associated server jobs are named QZDASOINIT, QZDASSINIT, and QTFPJTCP.

***DTAQ**
The data queue server daemon is started in the QSYSWRK subsystem. The daemon job is named QZHQSRVD. The associated server job is named QZHQSSRV.

***FILE** The file server daemon is started in the QSERVER subsystem. The daemon job is named QPWFSERVSD. The associated server jobs are named QPWFSERVSO, QPWFSERVSS, and QPWFSERVS2.

***NETPRT**
The network print server daemon is started in the QSYSWRK subsystem. The daemon job is named QNPSERVD. The associated server jobs are named QNPSERVS and QIWVPPJT.

***RMTCMD**
The remote command and distributed program call server daemon is started in the QSYSWRK subsystem. The daemon job is named QZRCSRVSD. The associated server job is named QZRCSRVS.

***SIGNON**
The signon server daemon is started in the QSYSWRK subsystem. The daemon job is named QZSOSGND. The associated server job is named QZSOSIGN.

***SVRMAP**
The server mapper daemon is started in the QSYSWRK subsystem. The daemon job is named QZSOSMAPD.

Top

Required protocol (RQDPCL)

Specifies which communication protocols are required to be active for the host server daemons to start.

Single Values

***ANY:** The TCP/IP communication protocol must be active at the time the STRHOSTSVR command is issued. If TCP/IP is not active, escape message PWS300D will be issued and the host server daemons will not be started. A diagnostic message (PWS3008) will be issued if TCP/IP is found to be inactive.

***NONE:**
No communication protocols need to be active at the time the STRHOSTSVR command is issued for the host server daemons to start. No messages will be issued for protocols which are inactive.

***TCP:** The TCP/IP communication protocol must be active at the time the STRHOSTSVR command is issued. If TCP/IP is not active, diagnostic message PWS3008 and escape message PWS300D will be issued and the host server daemons will not be started.

Top

Examples

None

Top

Error messages

***ESCAPE Messages**

PWS300D
Unable to start host server daemon jobs.

PWS3006
Errors occurred starting server daemon jobs.

Top

Start IDDU (STRIDD)

Where allowed to run: Interactive environments (*INTERACT *IPGM *IREXX *EXEC)
Threadsafe: No

Parameters
Examples
Error messages

The Start Interactive Data Definition Utility (STRIDD) command runs the main Interactive Data Definition (IDDU) menu. From this menu, you can select from options that allow you to work with data definitions, data dictionaries, files, and libraries, or use related commands and office tasks.

Top

Parameters

None

Top

Examples

```
STRIDD
```

This commands displays the main IDDU menu.

Top

Error messages

None

Top

Start IP over SNA Interface (STRIPSIFC)

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

Parameters
Examples
Error messages

The Start IP over SNA Interface (STRIPSIFC) command is used to start an AF_INET sockets over SNA interface (an IP address by which this local host is known on the SNA transport).

Restriction: Only eight (8) AF_INET sockets over SNA interfaces can be active on a single host. If the maximum number of interfaces is active and you want to start another interface, you must first end one or more interfaces using the End IP over SNA Interfaces (ENDIPSIFC) CL command.

Top

Parameters

Keyword	Description	Choices	Notes
INTNETADR	Internet address	*Character value*	Required, Positional 1

Top

Internet address (INTNETADR)

Specifies the internet address of an inactive (ended) interface that had previously been added to the IP over SNA configuration with the ADDIPSIFC CL command. The internet address is specified in the form *nnn.nnn.nnn.nnn*, where *nnn* is a decimal number ranging from 0 through 255. If the internet address is entered from a command line, the address must be enclosed in apostrophes.

This is a required parameter.

Top

Examples

```
STRIPSIFC   INTNETADR('9.5.1.248')
```

This command activates (starts) the interface with IP address 9.5.1.248.

Top

Error messages

***ESCAPE Messages**

CPFA10F
IP over SNA interface &1 not started.

Top

Start ITF (STRITF)

Where allowed to run: Interactive environments (*INTERACT *IPGM *IREXX *EXEC)
Threadsafe: No

Parameters
Examples
Error messages

The Start Interactive Terminal Facility (STRITF) command allows the user to send and receive data and file members for 5250 work stations using asynchronous communications. You can also send documents using the Interactive Terminal Facility (ITF). Before you can use ITF, you must start asynchronous communications.

Top

Parameters

Keyword	Description	Choices	Notes
RMTLOCNAME	Remote location	*Communications name*	Required, Positional 1

Top

Remote location (RMTLOCNAME)

Specifies the name of the remote location with which you want to communicate. This name is the same as the remote location name specified during configuration.

This is a required parameter.

Top

Examples

```
STRITF   CHICAGO
```

This command allows the user to communicate with the remote location CHICAGO.

Top

Error messages

None

Top

Start Journal (STRJRN)

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

Parameters
Examples
Error messages

The Start Journal (STRJRN) command is used to start journaling changes (made to an object or list of objects) to a specific journal. The object types which are supported through this interface are Data Areas (*DTAARA), Data Queues (*DTAQ), Stream Files (*STMF), Directories (*DIR), and Symbolic Links (*SYMLNK). Only objects of type *STMF, *DIR or *SYMLNK that are in the ″root″ (/), QOpenSys, and user-defined file systems are supported. For more information about the possible journal entries which can be sent, see the Journal management topic collection in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/. Search for ″journal entry finder″.

The user can specify that only the after image or both the before and the after images of an object of type *DTAARA be journaled. Before images are necessary to remove journaled changes using the Remove Journaled Changes (RMVJRNCHG) command.

After journaling begins for the object, the user should save the journaled object to preserve its journal attribute information. Also, the object must be saved because, for example, journaled changes cannot be applied to a version of the object that was saved before journaling was in effect.

For other ways to start journaling see the following commands:

- Access Paths - Start Journal Access Path (STRJRNAP)
- Physical Files - Start Journal Physical File (STRJRNPF)
- Libraries - Start Journal Library (STRJRNLIB)
- Other Objects - Start Journal Object (STRJRNOBJ)

Restrictions:

- The object must not be journaling changes to another journal.
- The maximum number of objects that can be associated with one journal is either 250,000 or 10,000,000. To get 10,000,000, the value of *MAX10M must have been specified for the JRNOBJLMT parameter on either the Create Journal (CRTJRN) command or on the Change Journal (CHGJRN) command. Once the number of objects is greater than or equal to this maximum, journaling does not start for any more objects.
- The specified journal must be a local journal. Although all object types which can be journaled to a local journal can also have their changes sent to a remote journal, this is accomplished by a two step process. First start journaling to the local journal. Then connect the local journal to a remote instance. To initiate such a connection, use the Add Remote Journal (ADDRMTJRN) command or the Add Remote Journal (QjoAddRemoteJournal) API. For information about remote journaling, see the Journal management topic collection in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/.
- The specified journal and object must reside in the same auxiliary storage pool (ASP).
- Stream files that are currently memory mapped, are virtual volume files, or are being used as IXS network storage spaces cannot be journaled.
- Objects that are internally marked as not eligible for journaling cannot be journaled. The system may mark system working directories that are created inside of user directories as not eligible for journaling.
- For data areas, only local external data area objects may be journaled. The special data areas (*LDA, *GDA, *PDA) and DDM data areas cannot be journaled.

- For data queues, only local data queues are supported. DDM data queues cannot be journaled.
- At least one of parameter OBJ or OBJFID must be specified.

Top

Parameters

Keyword	Description	Choices	Notes
OBJ	Objects	Values (up to 300 repetitions): *Element list*	Optional
	Element 1: Name	*Path name*	
	Element 2: Include or omit	***INCLUDE**, *OMIT	
OBJFID	File identifier	Values (up to 300 repetitions): *Hexadecimal value*	Optional
JRN	Journal	*Path name*	Optional
SUBTREE	Directory subtree	***NONE**, *ALL	Optional
PATTERN	Name pattern	Values (up to 20 repetitions): *Element list*	Optional
	Element 1: Pattern	*Character value*, *	
	Element 2: Include or omit	***INCLUDE**, *OMIT	
INHERIT	New objects inherit journaling	***NO**, *YES	Optional
IMAGES	Images	***AFTER**, *BOTH	Optional
OMTJRNE	Omit journal entry	***NONE**, *OPNCLOSYN	Optional
LOGLVL	Logging level	***ERRORS**, *ALL	Optional

Top

Objects (OBJ)

Specifies a maximum of 300 objects for which changes are to be journaled. Only objects whose path name identifies an object of type *STMF, *DIR, *SYMLNK, *DTAARA or *DTAQ are supported.

Element 1: Name

'object-path-name'

Specify the path name of the object for which changes are to be journaled.

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. Symbolic links within the path name will not be followed. If the path name begins with the tilde character, then the path is assumed to be relative to the appropriate home directory.

Additional information about path name patterns is in the Integrated file system topic collection in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/.

Note: This parameter is Unicode-enabled. See "Unicode support in CL" in the CL topic collection in the Programming category in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/ for additional information.

Element 2: Include or omit

The second element specifies whether names that match the pattern should be included or omitted from the operation. Note that in determining whether a name matches a pattern, relative name patterns are always treated as relative to the current working directory.

***INCLUDE**
The objects that match the object name pattern are to be journaled, unless overridden by an *OMIT specification.

***OMIT**
The objects that match the object name pattern are not be journaled. This overrides an *INCLUDE specification and is intended to be used to omit a subset of a previously selected path.

Top

File identifier (OBJFID)

Specifies a maximum of 300 file identifiers (FID) for which changes are to be journaled. FIDs are a unique identifier associated with integrated file system related objects. This field is input in hexadecimal format. Only objects whose FID identifies an object of type *STMF, *DIR, or *SYMLNK that is in the ″root″ (/), QOpenSys, or user-defined file systems, or objects of type *DTAARA or *DTAQ are supported.

file-identifier
Objects identified with the FID are journaled.

Top

Journal (JRN)

Specifies the path name of the journal that receives the journaled changes.

Note: This parameter is Unicode-enabled. See ″Unicode support in CL″ in the CL topic collection in the Programming category in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/ for additional information.

'journal-path-name'
Specify the path name of the journal that receives the journaled changes.

Top

Directory subtree (SUBTREE)

Specifies whether the directory subtrees are included in the start journal operation.

Note: This parameter is ignored if the OBJ parameter is not specified.

Note: This parameter is ignored unless object-path-name is a directory (*DIR) object.

***NONE**
Only the objects that match the selection criteria are processed. The objects within selected directories are not implicitly processed.

***ALL** All objects that meet the selection criteria are processed in addition to the entire subtree of each directory that matches the selection criteria. The subtree includes all subdirectories and the objects within those subdirectories.

Once the command has begun processing a specific directory subtree, the objects which will be found and processed may be affected by operations that update the organization of objects within the specified directory tree. This includes, but is not limited to, the following:

- Adding, removing, or renaming object links
- Mounting or unmounting file systems
- Updating the effective root directory for the process calling the command

- Updating the contents of a symbolic link

In order to process the directory subtree, the system code may increase the process-scoped maximum number of file descriptors that can be opened during processing. This is done so that the command is not likely to fail due to a lack of descriptors. This process-scoped maximum value is not reset when the command completes.

Top

Name pattern (PATTERN)

Specifies a maximum of 20 patterns to be used to include or omit objects for the start journal operation.

Only the last part of the path name will be considered for the name pattern match. Path name delimiters are not allowed in the name pattern. 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. Symbolic links within the path name will not be followed.

If this parameter is not specified, the default will be to match all patterns.

Additional information about path name patterns is in the Integrated file system topic collection in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/.

Note: This parameter is ignored if the OBJ parameter is not specified.

Note: This parameter applies to objects that exist when the start journal command is processed. This parameter does not apply to objects that will be created later in a journaled directory where new objects inherit journaling.

Element 1: Pattern

′*′
All objects that match the input OBJ parameter are to be included into the start journal operation or omitted from the start journal operation.

name-pattern
Specify the pattern to either include or omit objects for the start journal operation.

Note: This parameter is Unicode-enabled. See ″Unicode support in CL″ in the CL topic collection in the Programming category in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/ for additional information.

Element 2: Include or omit

The second element specifies whether names that match the pattern should be included or omitted from the operation. Note that in determining whether a name matches a pattern, relative name patterns are always treated as relative to the current working directory.

***INCLUDE**
The objects that match the object name pattern are included into the start journal operation unless overridden by an *OMIT specification.

***OMIT**
The objects that match the object name pattern are not to be included into the start journal operation. This overrides an *INCLUDE specification and is intended to be used to omit a subset of a previously selected pattern.

Top

New objects inherit journaling (INHERIT)

Specifies whether new objects created within a journaled directory should inherit the journal options and the journal state of its parent directory.

***NO** New objects created within the directory will not inherit the journal options and journal state of the parent directory.

***YES** New objects created within the directory will inherit the journal options and journal state of the parent directory.

Top

Images (IMAGES)

Specifies the kinds of images that are written to the journal receiver for changes to objects.

***AFTER**
Only *after* images are generated for changes to objects.

***BOTH**
The system generates both *before* and *after* images for changes to objects.

Note: The value *BOTH is only valid for *DTAARA objects.

Top

Omit journal entry (OMTJRNE)

Specifies the journal entries that are omitted.

***NONE**
No entries are omitted.

***OPNCLOSYN**
Open, close and force entries are omitted. Open, close and force operations on the specified objects do not generate open, close and force journal entries. This prevents the use of TOJOBO and TOJOBC entries on the Apply Journaled Changes (APYJRNCHG) command, but it saves some storage space in the journal receivers.

Note: The value *OPNCLOSYN is only valid for *DIR and *STMF objects.

Top

Logging level (LOGLVL)

Specifies the error logging level used. This parameter is used to determine which messages will be sent.

***ERRORS**
All diagnostic and escape messages are sent but the command will not send successful completion messages for each object. At the completion of this command, one completion message will be sent.

***ALL** The command sends all the messages that would be sent with *ERRORS and it will also send the successful completion message for each object.

Top

Examples

Example 1: Start Journaling with Omit of Directory

```
STRJRN   OBJ(('/mypath' *INCLUDE)
             ('/mypath/myobject' *OMIT))
         JRN('/QSYS.LIB/MYLIB.LIB/JRNLA.JRN')
```

This command journals all changes to all objects supported by this command within the first-level of directory '/mypath' except '/mypath/myobject' to journal '/QSYS.LIB/MYLIB.LIB/JRNLA.JRN'. None of the objects within the subdirectories of '/mypath' will be journaled.

Only the *after* images of updated records are written to the journal.

Example 2: Start Journaling with Pattern Matching

```
STRJRN   OBJ(('/mypath' *INCLUDE)
             ('/mypath/myobject.txt' *OMIT))
         JRN('/QSYS.LIB/MYLIB.LIB/JRNLA.JRN')  SUBTREE(*ALL)
         PATTERN(('*.TXT' *INCLUDE))  OMTJRNE(*OPNCLOSYN)
```

This command journals changes to all objects that match pattern '*.txt' in directory '/mypath' except object '/mypath/myobject.txt'. The open, close and force entries are not journaled.

Only the *after* images of updated records are written to the journal.

Example 3: Start Journaling with Omit by Pattern

```
STRJRN   OBJ(('/mypath/my*' *INCLUDE))
         JRN('/QSYS.LIB/MYLIB.LIB/JRNLA.JRN')
         PATTERN(('*.DTA*' *OMIT))
```

This command journals changes to all objects within the first-level directories that match the pattern for path '/mypath/my*' and will omit all objects that match pattern '*.DTA*' (objects of type *DTAARA and *DTAQ).

Only the *after* images of updated records are written to the journal.

Example 4: Start Journaling using File Identifiers

```
STRJRN   OBJFID(00000000000000007E09BDB000000009
                00000000000000009E09BDB00000000A)
         JRN('/QSYS.LIB/MYLIB.LIB/JRNLA.JRN')
```

This command journals all changes to the objects represented by the specified file identifiers to journal '/QSYS.LIB/MYLIB.LIB/JRNLA.JRN'.

Only the *after* images of updated records are written to the journal.

Example 5: Start Journaling on a Set of Data Queues

```
STRJRN   OBJ(('/QSYS.LIB/MYLIB.LIB/MYDATA*.DTAQ'))
         JRN('/QSYS.LIB/MYLIB.LIB/MYJRN.JRN')
```

This command starts the journaling of all changes to the objects of type *DTAQ in library MYLIB that begin with the characters ’MYDATA’.

Top

Error messages

***ESCAPE Messages**

CPFA0D4
File system error occurred. Error number &1.

CPF700A
&1 of &2 objects have started journaling.

CPF705A
Operation failed due to remote journal.

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.

CPF9825
Not authorized to device &1.

CPF9830
Cannot assign library &1.

CPF9873
ASP status is preventing access to object.

CPF9875
Resources exceeded on ASP &1.

Top

Start Journal Access Path (STRJRNAP)

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

Parameters
Examples
Error messages

The Start Journal Access Path (STRJRNAP) command is used to start journaling the access paths for all members of a database file to a specified journal. Any new member that is later added to the file also has its access path journaled.

If a physical file is specified, journaling can be started for its access paths. When access path journaling is started for a physical file, only the access paths for the physical file members are journaled. Journaling for any logical file access paths is started only when access path journaling is started for the logical file.

The journal entries created after running this command cannot be used in any apply or remove journaled changes operation. These entries are used only to recover the access path without rebuilding it after an abnormal system operation ending.

If you start journaling your access paths, consider specifying RCVSIZOPT(*RMVINTENT) on either the Create Journal (CRTJRN) or the Change Journal (CHGJRN) command for this journal. This will reduce the additional storage required to do access path journaling.

If you do not want the overhead of managing the access path journaling yourself, consider taking advantage of the system-managed access-path protection support. For more information, see the Journal management topic collection in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/, and the Edit Recovery for Access Paths (EDTRCYAP) or the Change Recovery for Access Paths (CHGRCYAP) command.

For other ways to start journaling see the following commands:

- Integrated file system objects - Start Journal (STRJRN)
- Physical files - Start Journal Physical File (STRJRNPF)
- Libraries - Start Journal Library (STRJRNLIB)
- Other objects - Start Journal Object (STRJRNOBJ)

Restrictions:

- Before journaling an access path, all physical files over which the access path is built must first be journaled to the same journal that is used to journal the access path. Even if all physical file members for a particular physical file are removed from the access path of a logical file, all physical files must still be journaled to the same journal before journaling the access path.
- The maximum number of objects that can be associated with one journal is either 250,000 or 10,000,000. To get 10,000,000, the value of *MAX10M must have been specified for the JRNOBJLMT parameter on either the Create Journal (CRTJRN) command or on the Change Journal (CHGJRN) command. Once the number of objects is greater than or equal to this maximum, journaling does not start for any more objects.
- All access paths to be journaled must specify MAINT(*IMMED) or MAINT(*DLY).
- If only after images are being journaled for the physical file members, the system automatically starts journaling the before and after images for the physical file once journaling is started for any access path built over the physical file. When journaling ends for the access paths, the system automatically stops journaling the before images for the physical file and again only journals the after images.
- Overrides are not applied to files specified on the FILE parameter.

- The specified journal must be a local journal. Although all object types which can be journaled to a local journal can also have their changes sent to a remote journal, this is accomplished by a two step process. First start journaling to the local journal. Then connect the local journal to a remote instance. To initiate such a connection, use the Add Remote Journal (ADDRMTJRN) command or the Add Remote Journal (QjoAddRemoteJournal) API. For information about remote journaling, see the Journal management topic collection in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/.
- Lock Processing

  The file with the access path to start journaling is locked with a read exclusive lock (*EXCLRD).

Top

Parameters

Keyword	Description	Choices	Notes
FILE	Journaled file	Values (up to 300 repetitions): *Qualified object name*	Required, Positional 1
	Qualifier 1: Journaled file	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
JRN	Journal	*Qualified object name*	Required, Positional 2
	Qualifier 1: Journal	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
LOGLVL	Logging level	***ERRORS**, *ALL	Optional

Top

Journaled file (FILE)

Specifies a maximum of 300 database files whose access paths are journaled.

This is a required parameter.

Qualifier 1: Journaled file

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

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

Top

Journal (JRN)

Specifies the journal that receives the file change journal entries.

This is a required parameter.

Qualifier 1: Journal

journal-name
Specify the name of the journal.

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.

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

Top

Logging level (LOGLVL)

Specifies the error logging level used. This parameter is used to determine which messages will be sent.

***ERRORS**
All diagnostic and escape messages are sent but the command will not send successful completion messages for each object. At the completion of this command, one completion message will be sent.

***ALL** The command sends all the messages that would be sent with *ERRORS and it will also send the successful completion message for each object.

Top

Examples

```
STRJRNAP   FILE(MYFILE)  JRN(MYLIB/JRNLA)
```

This command journals all access paths for all members in file MYFILE (found using the library search list) to journal JRNLA in library MYLIB.

Top

Error messages

***ESCAPE Messages**

CPF6971
Damage prevents object &1 from being journaled.

CPF6972
Cannot allocate access path for file &1 in &2.

CPF7003
Entry not journaled to journal &1. Reason code &3.

CPF7004
Maximum number of objects journaled to journal &1.

CPF7008
Cannot start or end access path journaling for file &1.

CPF7009
Not all based-on files being journaled to &3.

CPF7011
Not enough storage or resources.

CPF703C
DDL transaction prevents journaling operation.

CPF703D
DDL transaction prevents journaling operation.

CPF703E
DDL transaction prevents journaling operation.

CPF7030
Object of type *&3 already being journaled.

CPF7031
Cannot allocate member &3 file &1 in &2.

CPF7033
Start or end journaling failed for member &3.

CPF7034
Logical damage of file &1 in &2.

CPF7035
Object &1 in &2 already known to journal.

CPF705A
Operation failed due to remote journal.

CPF7079
Access path journaling for file &1 not started.

CPF708D
Journal receiver found logically damaged.

CPF7084
Object of type *&6 could not be journaled.

CPF709D
Cannot start journaling object of type *&7.

CPF9801
Object &2 in library &3 not found.

CPF9802
Not authorized to object &2 in &3.

CPF9803
Cannot allocate object &2 in library &3.

CPF9812
File &1 in library &2 not found.

CPF9820
Not authorized to use library &1.

CPF9822
Not authorized to file &1 in library &2.

CPF9825
Not authorized to device &1.

CPF9830
Cannot assign library &1.

CPF9873
ASP status is preventing access to object.

CPF9875
Resources exceeded on ASP &1.

Top

Start Journal Library (STRJRNLIB)

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

Parameters
Examples
Error messages

The Start Journal Library (STRJRNLIB) command is used to start journaling changes (made to a library or list of libraries) to a specific journal, and optionally to start journaling changes to objects within the library or list of libraries. Objects created in, moved into, or restored into a journaled library may be automatically journaled to the same journal the library is journaled to. For more information about journal entries which can be sent, see the Journal management topic collection in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/. Search for ″journal entry finder″.

After journaling begins for the object, the user should save the journaled object to preserve its journal attribute information. Also, the object must be saved because, for example, journaled changes cannot be applied to a version of the object that was saved before journaling was in effect.

Objects created, moved, or restored into the library that are eligible for journaling may automatically start journaling to the same journal as the library. Which objects inherit the journal state of the library and what journaling attributes they start journaling with are determined by the inherit journaling attributes of the library.

Use the Display Library Description (DSPLIBD) command to display journal information including the inherit rules for the library.

For other ways to start journaling see the following commands:

- Access paths - Start Journal Access Path (STRJRNAP)
- Integrated file system objects - Start Journal (STRJRN)
- Physical files - Start Journal Physical File (STRJRNPF)
- Other objects - Start Journal Object (STRJRNOBJ)

Restrictions:

- The object must not be journaling changes to another journal.
- The maximum number of objects that can be associated with one journal is either 250,000 or 10,000,000. To get 10,000,000, the value of *MAX10M must have been specified for the JRNOBJLMT parameter on either the Create Journal (CRTJRN) command or on the Change Journal (CHGJRN) command. Once the number of objects is greater than or equal to this maximum, journaling does not start for any more objects.
- The specified journal must be a local journal. Although all object types which can be journaled to a local journal can also have their changes sent to a remote journal, this is accomplished by a two step process. First start journaling to the local journal. Then connect the local journal to a remote instance. To initiate such a connection, use the Add Remote Journal (ADDRMTJRN) command or the Add Remote Journal (QjoAddRemoteJournal) API. For information about remote journaling, see the Journal management topic collection in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/.
- The specified journal and object must reside in the same auxiliary storage pool (ASP).
- If the specified library contains a data area named QDFTJRN, no objects being created, moved, or restored into the library will inherit the journaling state of the library. Instead, objects will automatically start journaling based on the contents of the QDFTJRN data area. When the QDFTJRN data area is deleted, objects being created, moved, and restored into the library will inherit the journaling state of the library based on the inherit rules.

- The following libraries may not be journaled: QSYS, QSYSxxxxx, QSYS2, QSYS2xxxxx, QRECOVERY, QRCYxxxxx, QSPL, QSPLxxxxx, QRCL, QRCLxxxxx, QRPLOBJ, QRPLxxxxx, QGPL, QTEMP, SYSIBM, SYSIBxxxxx, SYSIBMADM, SYSPROC, and SYSTOOLS, where xxxxx is a five-digit number of an Independent Auxiliary Storage Pool (IASP).
- Lock Processing

  The library to start journaling is locked with a read exclusive lock (*EXCLRD).

Top

Parameters

Keyword	Description	Choices	Notes
LIB	Library	Values (up to 300 repetitions): *Generic name, name*	Required, Positional 1
JRN	Journal	*Qualified object name*	Required, Positional 2
	Qualifier 1: Journal	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
INHRULES	Inherit rules	Values (up to 20 repetitions): *Element list*	Optional
	Element 1: Object type	***ALL**, *FILE, *DTAARA, *DTAQ	
	Element 2: Operation	***ALLOPR**, *CREATE, *MOVE, *RESTORE, *RSTOVRJRN	
	Element 3: Rule action	***INCLUDE**, *OMIT	
	Element 4: Images	***OBJDFT**, *AFTER, *BOTH	
	Element 5: Omit journal entry	***OBJDFT**, *NONE, *OPNCLO	
LOGLVL	Logging level	***ERRORS**, *ALL	Optional

Top

Library (LIB)

Specifies a maximum of 300 library names for which changes are to be journaled.

generic-name
Specify the generic name of the library for which changes are to be journaled. A generic name is a character string that contains one or more characters followed by an asterisk (*). If a generic name is specified, all libraries that have names with the same prefix as the generic name will have their changes journaled.

library-name
Specify the name of the library for which changes are to be journaled.

Top

Journal (JRN)

Specifies the journal that receives the journaled changes.

This is a required parameter.

Qualifier 1: Journal

journal-name
Specify the name of the journal that receives the journaled changes.

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.

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

Top

Inherit rules (INHRULES)

Specifies which objects created within a journaled library, moved into a journaled library, or restored into a journaled library should inherit the journal state of the library and which journaling attributes those objects should have.

Each rule defines object types and operations that the rule applies to. Multiple rules can be defined for the same set of objects. If multiple rules are defined for the same object and operation, the last rule defined for that object will be applied.

Note: If a data area named QDFTJRN exists within the library, no objects created, moved, or restored into the library will inherit the journal state of the library. Objects created, moved, and restored into the library will automatically start journaling based on the data area information.

Element 1: Object type

Specify the object type of the objects that are identified by this rule.

***ALL** This rule applies to all object types that can be journaled.

***DTAARA**
This rule applies to data areas.

***DTAQ**
This rule applies to data queues.

***FILE** This rule applies to database physical files.

Element 2: Operation

Specifies the operation or operations identified by this rule.

***ALLOPR**
This rule applies to all objects created in, moved into, or restored into the library. This is a combination of the values *CREATE, *MOVE, and *RESTORE. See their descriptions for further information.

***CREATE**
This rule applies to all objects created in the library.

***MOVE**
This rule applies to all objects moved into the library, if they are not already journaled.

***RESTORE**
This rule applies to all objects restored into the library. If an object is restored over a currently existing object, the restored object will retain the same journal state and journaling attributes of the object it was restored over. If an object was never journaled when it was saved, journaling is started for the object when it is restored into the library. Otherwise, if an object was journaled

when it was saved, it will first attempt to start journaling to the journal it was journaled to when it was saved, with the same journaling attributes it had when it was saved. If that journal does not exist, the object will start journaling to the same journal the library is journaled to, with the journaling attributes defined by this rule. A rule with *RESTORE specified will override any previous rule for the same object type with *RSTOVRJRN specified.

If the accompanying rule action is set to *OMIT and the object is not restored over an existing object, the restored object will retain the same journal state and journaling attributes as it had when it was saved. The journal state of the library is ignored.

***RSTOVRJRN**

This rule applies to all objects restored into the library. If an object is restored over a currently existing object, the restored object will retain the same journal state and journaling attributes of the object it was restored over. Otherwise, the object will start journaling to the same journal the library is journaled to, with the journaling attributes defined by this rule. A rule with *RSTOVRJRN specified will override any previous rule for the same object type with *RESTORE specified.

If the accompanying rule action is set to *OMIT and the object is not restored over an existing object, the restored object will retain the same journal state and journaling attributes as it had when it was saved. The journal state of the library is ignored.

Element 3: Rule action

Specifies whether or not the objects that match object type and operation in this rule will be included or omitted from the list of objects that inherit the journal state of the library.

***INCLUDE**

All objects that match the object type and operation will inherit the journal state of the library, and will use the journaling attributes defined by this rule.

***OMIT**

All objects that match object type and operation will not inherit the journal state of the library. This overrides an *INCLUDE specification and is intended to be used to omit a subset of a previously defined rule.

Element 4: Images

Specifies the kinds of images that are written to the journal receiver for changes to objects for objects that inherit the journal state of the library.

Note: This element is ignored if *OMIT is specified for Rule action.

***OBJDFT**

The default value for each object type will be used for this journaling attribute when an object inherits the journal state of the library. Database files (*FILE) will have both *before* and *after* images generated by the system (*BOTH). All other object types will have only *after* images generated by the system (*AFTER).

***AFTER**

Only *after* images are generated for objects that inherit the journal state of the library.

***BOTH**

Both *before* and *after* images are generated for objects that inherit the journal state of the library.

Note: The value *BOTH is only valid for data area (*DTAARA) and database file (*FILE) objects. If *BOTH is specified and *ALL is specified for Object type, the system will generate both *before* and *after* images for data areas and database files. All other object types will only generate *after* images.

Element 5: Omit journal entry

Specifies the journal entries that are omitted for objects that inherit the journal state of the library.

Note: This element is ignored if *OMIT is specified for Rule action.

***OBJDFT**
The default value for each object type will be used for this journaling attribute when an object inherits the journal state of the library. Database files (*FILE) will omit open and close entries. No other object types will omit journal entries.

***NONE**
No journal entries will be omitted for objects that inherit the journal state of the library.

***OPNCLO**
Open and close entries are omitted for database file (*FILE) objects that inherit the journal state of the library. This prevents the use of TOJOBO and TOJOBC entries on the Apply Journaled Changes (APYJRNCHG) command, but it saves some storage space in the journal receivers.

Note: The value *OPNCLO is only valid for database files (*FILE). If *OPNCLO is specified and *ALL is specified for Object type, database files will omit open and close entries. All other object types will not omit any journal entries.

Top

Logging level (LOGLVL)

Specifies the error logging level used. This parameter is used to determine which messages will be sent.

***ERRORS**
All diagnostic and escape messages are sent but the command will not send successful completion messages for each object. At the completion of this command, one completion message will be sent.

***ALL** The command sends all the messages that would be sent with *ERRORS and it will also send the successful completion message for each object.

Top

Examples

Example 1: Start Journaling a Single Library

```
STRJRNLIB  LIB(MYLIBA) JRN(MYLIBA/JRN) JRNCUROBJ(*NONE)
```

This command journals all changes to library MYLIBA to journal JRN in MYLIBA. Also, all objects that can be journaled that are created in, moved into, or restored into this library will automatically start journaling to journal JRN in MYLIBA.

Example 2: Start Journaling a Library and Selectively Inherit Journaling

```
STRJRNLIB LIB(MYLIBC) JRN(MYLIBC/JRN)
          INHRULES((*FILE *ALLOPR *INCLUDE *BOTH *OPNCLOSYN)
                   (*DTAARA *CREATE *INCLUDE *OBJDFT *OBJDFT))
```

This command journals all changes to library MYLIBC to journal JRN in MYLIBC. All files that are created in, moved into, or restored into library MYLIBC will have journaling started to journal JRN in library MYLIBC, and will generate both before and after images and will omit open and close entries. All data areas that are created in library MYLIBC will have journaling started.

Example 3: Start Journaling a Library and Selectively Inherit Journaling with Omit

```
STRJRNLIB LIB(MYLIBD) JRN(MYLIBA/JRN)
          INHRULES((*ALL *ALLOPR *INCLUDE *OBJDFT *OBJDFT)
                   (*FILE *CREATE *OMIT)
                   (*FILE *RESTORE *OMIT))
```

This command journals all changes to library MYLIBD to journal JRN in MYLIBA. All objects that can be journaled that are created in, moved into, or restored into library MYLIBD will have journaling started to journal JRN in MYLIBA, except any files that are created or restored into the library.

Top

Error messages

***ESCAPE Messages**

CPF6979
Journal &1 in library &2 is unusable.

CPF700A
&1 of &2 objects have started journaling.

CPF705A
Operation failed due to remote journal.

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.

CPF9825
Not authorized to device &1.

CPF9830
Cannot assign library &1.

CPF9873
ASP status is preventing access to object.

CPF9875
Resources exceeded on ASP &1.

Top

Start Journal Object (STRJRNOBJ)

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

Parameters
Examples
Error messages

The Start Journal Object (STRJRNOBJ) command is used to start journaling changes (made to an object or list of objects) to a specific journal. The object types which are supported through this interface are Data Areas (*DTAARA) and Data Queues (*DTAQ). For more information about journal entries which can be sent, see the Journal management topic collection in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/. Search for ″journal entry finder″.

Additionally, the user can specify that only the after image or both the before and the after images of an object of type *DTAARA be journaled. Before images are necessary to remove journaled changes using the Remove Journaled Changes (RMVJRNCHG) command.

After journaling begins for the object, the user should save the journaled object to preserve its journal attribute information. Also, the object must be saved because, for example, journaled changes cannot be applied to a version of the object that was saved before journaling was in effect.

For other ways to start journaling see the following commands:
- Access paths - Start Journal Access Path (STRJRNAP)
- Integrated file system objects - Start Journal (STRJRN)
- Physical files - Start Journal Physical File (STRJRNPF)
- Libraries - Start Journal Library (STRJRNLIB)

Restrictions:
- The object must not be journaling changes to another journal.
- The maximum number of objects that can be associated with one journal is either 250,000 or 10,000,000. To get 10,000,000, the value of *MAX10M must have been specified for the JRNOBJLMT parameter on either the Create Journal (CRTJRN) command or on the Change Journal (CHGJRN) command. Once the number of objects is greater than or equal to this maximum, journaling does not start for any more objects.
- The specified journal must be a local journal. Although all object types which can be journaled to a local journal can also have their changes sent to a remote journal, this is accomplished by a two step process. First start journaling to the local journal. Then connect the local journal to a remote instance. To initiate such a connection, use the Add Remote Journal (ADDRMTJRN) command or the Add Remote Journal (QjoAddRemoteJournal) API. For information about remote journaling, see the Journal management topic collection in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/.
- The specified journal and object must reside in the same auxiliary storage pool (ASP).
- For data areas, only local external data area objects may be journaled. The special data areas (*LDA, *GDA, and *PDA) and DDM data areas cannot be journaled.
- For data queues, only local data queues are supported. DDM data queues cannot be journaled.
- Lock Processing

  The object to start journaling is locked with a read exclusive lock (*EXCLRD).

Top

Parameters

Keyword	Description	Choices	Notes
OBJ	Object	Values (up to 300 repetitions): *Qualified object name*	Required, Positional 1
	Qualifier 1: Object	*Generic name, name*, *ALL	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
OBJTYPE	Object type	*DTAARA, *DTAQ	Required, Positional 2
JRN	Journal	*Qualified object name*	Required, Positional 3
	Qualifier 1: Journal	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
IMAGES	Images	***AFTER**, *BOTH	Optional
LOGLVL	Logging level	***ERRORS**, *ALL	Optional

Top

Object (OBJ)

Specifies a maximum of 300 qualified object names for which changes are to be journaled.

This is a required parameter.

Qualifier 1: Object

***ALL** All objects of the specified type in the specified library will have their changes written to the journal.

generic-name
Specify the generic name of the object for which changes are to be journaled. A generic name is a character string that contains one or more characters followed by an asterisk (*). If a generic name is specified, all objects of the specified object type that have names with the same prefix as the generic name will have their changes journaled.

object-name
Specify the name of the object for which changes are to be journaled.

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.

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

Top

Object type (OBJTYPE)

Specifies the object type for which changes are to be journaled.

This is a required parameter.

***DTAARA**
Changes for data area objects are to be journaled.

***DTAQ**
Changes for data queue objects are to be journaled.

Top

Journal (JRN)

Specifies the journal that receives the journaled changes.

This is a required parameter.

Qualifier 1: Journal

journal-name
Specify the name of the journal that receives the journaled changes.

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.

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

Top

Images (IMAGES)

Specifies the kinds of images that are written to the journal receiver for changes to objects.

***AFTER**
Only *after* images are generated for changes to objects.

***BOTH**
The system generates both *before* and *after* images to the journal for changes to objects.

Note: The value *BOTH is only valid for *DTAARA objects.

Top

Logging level (LOGLVL)

Specifies the error logging level used. This parameter is used to determine which messages will be sent.

***ERRORS**
All diagnostic and escape messages are sent but the command will not send successful completion messages for each object. At the completion of this command, one completion message will be sent.

***ALL** The command sends all the messages that would be sent with *ERRORS and it will also send the successful completion message for each object.

Top

Examples

```
STRJRNOBJ   OBJ(DTALIB/MYDTAARA)  OBJTYPE(*DTAARA)
            JRN(MYLIB/JRNLA)
```

This command journals all changes to data area MYDTAARA in library DTALIB to journal JRNLA in library MYLIB. Only the *after* images of updates are written to the journal.

Top

Error messages

***ESCAPE Messages**

CPF6979
Journal &1 in library &2 is unusable.

CPF700A
&1 of &2 objects have started journaling.

CPF705A
Operation failed due to remote journal.

CPF7057
*LIBL not allowed with *ALL or *ALLLIB or a generic name.

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.

CPF9825
Not authorized to device &1.

CPF9830
Cannot assign library &1.

CPF9873
ASP status is preventing access to object.

CPF9875
Resources exceeded on ASP &1.

Top

Start Journal Physical File (STRJRNPF)

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

Parameters
Examples
Error messages

The Start Journal Physical File (STRJRNPF) command is used to start journaling changes made to a specific database physical file to a specific journal. Changes in new members added to the file are also journaled.

The user can specify that only the after image or both the before and after images of records in the journaled physical file be journaled. Before images are necessary to remove journaled changes using the Remove Journaled Changes (RMVJRNCHG) command. In addition, the system will automatically capture the before images for a database file if the file is opened under commitment control. For more information about commitment control, see the Database category in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/.

After journaling begins for the file, and after any new members are added to the file, the user should run the Save Changed Object (SAVCHGOBJ) command with OBJTYPE(*FILE) and OBJJRN(*YES) specified. The file must be saved because journaled changes cannot be applied to a version of the file that was saved before journaling was in effect.

When the file being journaled is a distributed file, the STRJRNPF command is also distributed if journaling was successfully started locally. Even if the distribution request fails, the local file remains journaled.

For other ways to start journaling see the following commands:

- Access paths - Start Journal Access Path (STRJRNAP)
- Integrated file system objects - Start Journal (STRJRN)
- Libraries - Start Journal Library (STRJRNLIB)
- Other objects - Start Journal Object (STRJRNOBJ)

Restrictions:

- The file must not be journaling changes to another journal.
- Overrides are not applied to files specified on the FILE parameter.
- The maximum number of objects that can be associated with one journal is either 250,000 or 10,000,000. To get 10,000,000, the value of *MAX10M must have been specified for the JRNOBJLMT parameter on either the Create Journal (CRTJRN) command or on the Change Journal (CHGJRN) command. Once the number of objects is greater than or equal to this maximum, journaling does not start for any more objects.
- The specified journal must be a local journal. Although all object types which can be journaled to a local journal can also have their changes sent to a remote journal, this is accomplished by a two step process. First start journaling to the local journal. Then connect the local journal to a remote instance. To initiate such a connection, use the Add Remote Journal (ADDRMTJRN) command or the Add Remote Journal (QjoAddRemoteJournal) API. For information about remote journaling, see the Journal management topic.
- In multithreaded jobs, this command is not threadsafe for distributed files and fails for distributed files that use relational databases of type *SNA.
- If the file has Large Object (LOB) columns, and the total of the LOB columns plus the record size is greater than 15,761,440 bytes, then the file can only be journaled to a journal with RCVSIZOPT(*MAXOPT2) or RCVSIZOPT(*MAXOPT3) specified.

- Lock Processing

  The physical file to start journaling is locked with a read exclusive lock (*EXCLRD). Any logical files built over the physical file are also locked with a read exclusive lock (*EXCLRD).

Top

Parameters

Keyword	Description	Choices	Notes
FILE	Physical file to be journaled	Values (up to 300 repetitions): *Qualified object name*	Required, Positional 1
	Qualifier 1: Physical file to be journaled	*Generic name*, *name*, *ALL	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
JRN	Journal	*Qualified object name*	Required, Positional 2
	Qualifier 1: Journal	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
IMAGES	Record images	***AFTER**, *BOTH	Optional
OMTJRNE	Journal entries to be omitted	***NONE**, *OPNCLO	Optional
LOGLVL	Logging level	***ERRORS**, *ALL	Optional

Top

Physical file to be journaled (FILE)

Specifies a maximum of 300 physical files whose changes are written to the journal.

This is a required parameter.

Qualifier 1: Physical file to be journaled

***ALL** All physical files in the specified library will have their changes written to the journal.

generic-name
Specify the generic name of the physical file for which changes are to be journaled. A generic name is a character string that contains one or more characters followed by an asterisk (*). If a generic name is specified, all physical files that have names with the same prefix as the generic name will have their changes journaled.

file-name
Specify the name of the physical file for which changes are to be journaled.

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.

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

Top

Journal (JRN)

Specifies the journal that will receive the file change journal entries.

This is a required parameter.

Qualifier 1: Journal

journal-name
Specify the name of the journal.

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.

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

Top

Record images (IMAGES)

Specifies the kinds of record images to be written to the journal for changes to records in the file.

*AFTER
Only *after* images are written to the journal for changes to records in this file.

*BOTH
The system writes both *before* and *after* images to the journal for changes to records in this file.

Top

Journal entries to be omitted (OMTJRNE)

Specifies the journal entries that are omitted.

*NONE
No journal entries are omitted.

*OPNCLO
Open and *close* entries are omitted. *Open* and *close* operations on the specified file members do not create *open* and *close* journal entries. This prevents the use of TOJOBO and TOJOBC entries on the Apply Journaled Changes (APYJRNCHG) and Remove Journaled Changes (RMVJRNCHG) commands, but it saves some storage space in the attached receivers.

Top

Logging level (LOGLVL)

Specifies the error logging level used. This parameter is used to determine which messages will be sent.

*ERRORS
All diagnostic and escape messages are sent but the command will not send successful completion messages for each object. At the completion of this command, one completion message will be sent.

***ALL** The command sends all the messages that would be sent with *ERRORS and it will also send the successful completion message for each object.

Top

Examples

```
STRJRNPF   FILE(MYFILE)  JRN(MYLIB/JRNLA)
```

This command journals all changes to all members of file MYFILE (as found using the library search list) to journal JRNLA in library MYLIB. Only the *after* images of updated records are written to the journal.

Top

Error messages

***ESCAPE Messages**

CPF6979
Journal &1 in library &2 is unusable.

CPF700A
&1 of &2 objects have started journaling.

CPF705A
Operation failed due to remote journal.

CPF7057
*LIBL not allowed with *ALL or *ALLLIB or a generic name.

CPF708D
Journal receiver found logically damaged.

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.

CPF9825
Not authorized to device &1.

CPF9830
Cannot assign library &1.

CPF9873
ASP status is preventing access to object.

CPF9875
Resources exceeded on ASP &1.

Top

Start Job Watcher (STRJW)

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

Parameters
Examples
Error messages

The Start Job Watcher (STRJW) command starts the collection of detailed job performance data. It allows the user to obtain detailed data related to a selected set of jobs or tasks on the system. Job Watcher collects data on a sampling basis and writes the collected data to a set of database files. The file names all begin with the letters 'QAPYJW'. In addition to basic job information the user may optionally request SQL, activation group, sockets and TCP, and call stack data.

Restrictions:

- To use this command, you must have service (*SERVICE) special authority, or be authorized to the Job Watcher function of the Operating System through System i5 Navigator's Application Administration support. The Change Function Usage (CHGFCNUSG) command, with a function ID of QIBM_SERVICE_JOB_WATCHER, can also be used to change the list of users that are allowed to use this command.
- You must have execute (*EXECUTE) authority to the library specified in the **Library (LIB)** parameter.

Top

Parameters

Keyword	Description	Choices	Notes
DFN	Definition	*Name*, ***SELECT**	Optional
COL	Collection	*Name*, ***GEN**	Optional
LIB	Library	*Name*, **QPFRDATA**	Optional
TEXT	Text 'description'	*Character value*, ***DFN**, *BLANK	Optional
RPLDTA	Replace data	***NO**, *YES	Optional
COLITV	Collection interval	0.1-3600.0, ***DFN**, *NODELAY	Optional
ENDCOL	End collection	Single values: *NOITVDTA Other values (up to 3 repetitions): *Element list*	Optional
	Element 1: Option	***NBRSEC**, *DASDMB, *NBRITV	
	Element 2: Value	*Integer*, **60**	
JOB	Job name	Single values: ***DFN**, *ALL, *NONE Other values (up to 20 repetitions): *Qualified job name*	Optional
	Qualifier 1: Job name	*Generic name, name*	
	Qualifier 2: User	*Generic name, name*, *ALL	
	Qualifier 3: Number	000000-999999, *ALL	
TASKNAME	Task name	Single values: ***DFN**, *ALL, *NONE Other values (up to 20 repetitions): *Character value*	Optional
TDENBR	TDE number	Single values: ***DFN** Other values (up to 20 repetitions): *Hexadecimal value*	Optional

Top

Definition (DFN)

Specifies the Job Watcher definition to use when starting a new Job Watcher collection. The definition identifies the parameters to use when starting a Job Watcher collection. A new Job Watcher definition can be created using the **Add Job Watcher Definition (ADDJWDFN)** command.

***SELECT**
The user will be prompted to select from a list of existing Job Watcher definitions. This value is not valid when the STRJW command is submitted to batch and will result in an error.

name Specify the name of the Job Watcher definition to use for this collection.

Top

Collection (COL)

Specifies the name of the collection. The name of the collection is the name of the database file member to which the output data will be written. If a member by this name does not exist in each Job Watcher database file, one will be created with the specified name. If a member by this name already exists, you must specify *YES on the **Replace data (RPLDTA)** parameter in order to write over the data in the existing member.

***GEN** The collection name will be generated based on when the STRJW command was called. The format of the collection name will be **Qdddhhmmss** where **ddd** is the Julian day (001-366) and **hhmmss** is the hour (01-24), minute (00-59) and second (00-59) the collection was started.

name Specify the name of the collection.

Top

Library (LIB)

Specifies the library where the database files for the Job Watcher data will exist. Each file that is not found in the specified library is automatically created in that library.

QPFRDATA
The database files will be located in library QPFRDATA.

name Specify the name of the library where the database files will be located.

Top

Text 'description' (TEXT)

Specifies the text to be used for each member across the set of Job Watcher database files associated with the collection.

***DFN** Job Watcher database file members will have the same text description as the Job Watcher definition that was selected on the **Definition (DFN)** parameter.

***BLANK**
Job Watcher database file members will have no text description.

character-value
Specify a text description for this set of Job Watcher database file members. The description should be no more that 50 characters of text, enclosed in apostrophes.

Top

Replace data (RPLDTA)

Specifies whether an existing set of Job Watcher database file members should be replaced with a new set of members with the same name.

***NO** The existing Job Watcher database file members will not be replaced. A new Job Watcher collection will not begin if the specified member already exists.

***YES** The existing Job Watcher database file members will be replaced with data from the new collection.

Top

Collection interval (COLITV)

Specifies the interval between retrieval of job/task data. Job/task data is collected from the system on a sampling basis. This value specifies the amount of time that will elapse between the collection of each sample.

***DFN** The collection interval specified in the Job Watcher definition will be used for this collection.

***NODELAY**
Data will be collected as fast as possible, with no delay between the collection of interval data.

0.1-3600.0
Specify the number of seconds to delay between the collection of interval data. If a value is specified on this parameter it will override the value in the Job Watcher definition.

Top

End collection (ENDCOL)

Specifies the criteria which will end the collection. This parameter allows you to specify multiple conditions which will end the collection of data. If more than one end option is specified, the Job Watcher collection will end the first time any one of the specified criteria has been met.

You can specify 3 values for this parameter.

Element 1: Option

Single values

***NOITVDTA**
End the collection after the first interval in which no records were collected.

Other values

***NBRSEC**
End the collection after a number of seconds has elapsed. Specify the time limit in element 2 of this parameter.

***DASDMB**
End the collection when a number of megabytes of data has been written to the Job Watcher database files. Specify the megabyte limit in element 2 of this parameter.

***NBRITV**
End the collection when a number of collection intervals has occurred. Specify the number of intervals in element 2 of this parameter.

Element 2: Value

60 If the default of *NBRSEC is used for element 1 of this parameter, the collection will end after sixty seconds of data is collected.

integer
Specify the number of seconds (for *NBRSEC) or the number of megabytes (for *DASDMB) or the number of intervals (for *NBRITV) to use as the ending criteria for the collection.

Top

Job name (JOB)

Specifies the jobs that will be included in the Job Watcher collection. If a value is specified on this parameter it will override the value in the Job Watcher definition.

Single values

***DFN** The jobs specified in the Job Watcher definition will be used for this collection.

***ALL** All jobs on the system are included.

***NONE**
None of the jobs on the system are included.

Other values (up to 20 repetitions)

Qualifier 1: Job name

name Specify the name of the job to include in the Job Watcher collection.

generic-name
Specify the generic name of the job to be included. 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. For more information about generic object names, see the CL topic collection in the Programming category in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/

Qualifier 2: User

***ALL** All jobs that match the specified job name are included.

name Specify the name of the user of the job to be included.

generic-name
Specify the generic user name of the jobs to be included.

Qualifier 3: Number

***ALL** All jobs that match the specified job name and user name are included.

number
Specify the job number to further qualify the job name and user name.

Top

Task name (TASKNAME)

Specifies the name of the task(s) which will be included in the Job Watcher collection. If a value is specified on this parameter it will override the value in the Job Watcher definition.

Single values

***DFN** The tasks specified in the Job Watcher definition will be used for this collection.

***ALL** All tasks on the system will be included in the collection.

***NONE**
None of the tasks on the system will be included in the collection.

Other values (up to 20 repetitions)

name Specify the name of the tasks which will be included in the collection.

Top

TDE number (TDENBR)

Specifies the Task Dispatching Element (TDE) number of the TDE(s) which will be included in the Job Watcher collection. The TDE number is a unique identifier assigned to each job, thread, and task running in the system. The TDE number may be found by using the Display list of tasks option in the Display/Alter/Dump function of Start System Service Tools (STRSST). If a value is specified on this parameter it will override the value in the Job Watcher definition.

Single values

***DFN** The TDE numbers specified in the Job Watcher definition will be used for this collection.

Other values (up to 20 repetitions)

number
The TDE number of the TDEs which will be included in the collection.

Top

Examples

Example 1: Start Job Watcher Using a Definition

```
STRJW   DFN(MYDFN)  COL(TEST)  LIB(MYLIB)
```

This command will start a Job Watcher collection using the definition called MYDFN to determine which data will be collected. Assuming the shipped default of ENDCOL(*NBRSEC 60) is used, data collection will end after 60 seconds. Collected data will be written to the Job Watcher database files in library MYLIB in member TEST. The Job Watcher database file names all begin with 'QAPYJW'.

Example 2: Start Job Watcher Using a Collection Interval

```
STRJW   DFN(MYDFN)  COLITV(5)  ENDCOL((*NBRITV 200))
```

This command will start Job Watcher to collect data using the definition called MYDFN. Data will be collected at an interval of 5 seconds, overriding the interval specified in the definition. Data collection will end after 200 intervals have been collected. Data will be written to the Job Watcher database files in library QPFRDATA in a member name which will be generated by Job Watcher based on the date and time the collection started.

Example 3: Start Job Watcher, Prompting to Select a Definition

```
STRJW   DFN(*SELECT)  COL(TEST)  LIB(MYLIB)
        ENDCOL((*DASDMB 100))
```

This command will open a prompt panel listing all Job Watcher definitions which currently exist on the system. The definition selected on this panel will be used to determine what data will be collected. Data collection will end after 100 megabytes of data have been written to the Job Watcher database files in library MYLIB in member TEST.

Top

Error messages

***ESCAPE Messages**

CPF2401
Not authorized to library &1.

CPF9810
Library &1 not found.

CPFB514
Member already exists.

CPFB517
Unexpected error in Job Watcher.

CPFB518
The user does not have the required authority.

CPFB519
Error detected in Job Watcher condition control file.

CPFB51A
Start Job Watcher did not complete successfully.

CPFB51B
Option *SELECT is only valid if the command is being run interactively.

Top

Start Job Log Server (STRLOGSVR)

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

Parameters
Examples
Error messages

The Start Job Log Server (STRLOGSVR) command is used to start the job log server which writes job logs for jobs that are in a job log pending state. See Job log output (LOGOUTPUT) for additional information on which jobs are handled by the server.

The job log server will write a job's job log either to a spooled file, to a printer, or to an outfile, if specified to do so (by using the QMHCTLJL, Control job log API).

Restrictions:

- You must have job control (*JOBCTL) special authority to use this command.

Top

Parameters

Keyword	Description	Choices	Notes
NBRSVR	Number of servers	1-30, ***CALC**	Optional, Positional 1

Top

Number of servers (NBRSVR)

Specifies the number of job log server jobs to be started. If one or more job log server jobs are already active, this is the number of additional server jobs that will be started. If the number of servers requested would exceed the maximum active, only the difference between the maximum and the current number of active servers will be started. The maximum number of job log servers that can be active at one time is 30.

***CALC**
The number of job log server jobs started will be determined by the system.

1-30 Specify the number of job log servers to start.

Top

Examples

```
STRLOGSVR   NBRSVR(*CALC)
```

This command starts the job log server. The number of server jobs started will be determined by the system.

Top

Error messages

***ESCAPE Messages**

CPF134A

Job log server not started.

Top

Start Mode (STRMOD)

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

Parameters
Examples
Error messages

The Start Mode (STRMOD) command starts one or all modes currently in use for an advanced program-to-program communications (APPC) remote location. The user can use STRMOD in either the reset or ended state; it is required only after an End Mode (ENDMOD) command has ended a mode. More information is in the APPC Programming book, SC41-5443.

Restriction: The user must have operational authority for the APPC device to use this command.

Top

Parameters

Keyword	Description	Choices	Notes
RMTLOCNAME	Remote location	*Communications name*	Required, Positional 1
DEV	Device	*Name*, ***LOC**	Optional, Positional 2
MODE	Mode	*Communications name*, ***NETATR**, *ALL	Optional
LCLLOCNAME	Local location	*Communications name*, ***LOC**, *NETATR	Optional
RMTNETID	Remote network identifier	*Communications name*, ***LOC**, *NETATR, *NONE	Optional

Top

Remote location (RMTLOCNAME)

Specifies the remote location name.

This is a required parameter.

Top

Device (DEV)

Specifies the device description name.

The possible values are:

***LOC** The device description is determined by the system.

device-name
Specify the name of the device description.

Top

Mode (MODE)

Specifies the mode that is to be started.

The possible values are:

***NETATR**
The mode in the network attributes is used.

***ALL** All modes currently in use for the remote location are to be started.

- For a device description automatically created by the APPN support or a device description manually created with the APPN parameter specified as *YES, *ALL indicates that any modes that have been used while the remote location was active, but are not currently started, are to be started.
- For a device description manually created with the APPN parameter specified as *NO, *ALL specifies that all configured modes for the specified remote location are to be started.

BLANK
The mode name (consisting of 8 blank characters) is used.

mode-name
Specify a mode name.

Note: SNASVCMG and CPSVCMG are reserved names and cannot be specified.

Top

Local location (LCLLOCNAME)

Specifies the local location name.

The possible values are:

***LOC** The local location name is determined by the system.

***NETATR**
The LCLLOCNAME value specified in the system network attributes is used.

local-location-name
Specify the name of your location. The local location name is specified if you want to indicate a specific local location name for the remote location.

Top

Remote network identifier (RMTNETID)

Specifies the remote network ID used with the remote location.

The possible values are:

***LOC** The system selects the remote network ID.

***NETATR**
The remote network identifier specified in the network attributes is used.

***NONE**
No remote network identifier (ID) is used.

remote-network-id
Specify the name of the remote network ID.

Top

Examples

```
STRMOD   RMTLOCNAME(APPCRLOC)  DEV(APPCDEV)  MODE(APPCMODE)
         RMTNETID(CHICAGO)
```

This command starts a mode named APPCMODE for a remote location named APPCRLOC, a device named APPCDEV, and a remote network ID of CHICAGO.

Top

Error messages

***ESCAPE Messages**

CPF598B
The &1 command failed for one or more modes.

Top

Start Mail Server Framework (STRMSF)

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

Parameters
Examples
Error messages

The Start Mail Server Framework (STRMSF) command starts the mail server framework jobs in the system work subsystem (QSYSWRK).

Top

Parameters

Keyword	Description	Choices	Notes
MSGOPT	How to process mail messages	***RESUME**, *RESET, *CLEAR	Optional, Positional 1
NBRMSFJOB	Number of MSF jobs	1-99, 3	Optional, Positional 2

Top

How to process mail messages (MSGOPT)

Specifies how the mail server framework processes existing mail server framework messages.

The possible values are:

***RESUME**
All existing mail server framework messages continue processing from the point the mail server framework previously ended.

***RESET**
All existing mail server framework messages are processed as if they were just created.

***CLEAR**
All existing mail server framework messages are deleted. This option should only be used when a software error is reported with the mail server framework or its associated exit point programs.

Top

Number of MSF jobs (NBRMSFJOB)

Specifies the number of mail server framework jobs to start. This option allows concurrent processing of several mail server framework messages.

The possible values are:

3 Three jobs are started.

number-of-jobs
Specify the number of jobs you want handling mail server framework messages. The valid values range from 1 through 99.

Top

Examples

Example 1: Starting One Mail Server Framework Job

```
STRMSF   NBRMSFJOB(1)
```

This command starts one mail server framwork job in a normal manner, processing any mail server framework messages at the point at which processing was interrupted.

Example 2: Restarting Mail Server Framework Jobs

```
STRMSF   NBRMSFJOB(3)  MSGOPT(*RESET)
```

This command starts three mail server framework jobs and any mail server framework messages which were partially handled by previous mail server famework jobs are processed again from the beginning.

Top

Error messages

***ESCAPE Messages**

CPFAFAA
STRMSF did not complete successfully.

CPFAFAD
Mail Server Framework currently active.

CPFAFA0
Errors detected on MSF internal message index.

CPFAFA1
Errors detected on MSF internal message queue.

CPFAFFF
Internal system error in program &1.

Top

Start NFS Server (STRNFSSVR)

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

Parameters
Examples
Error messages

The Start Network File System Server (STRNFSSVR) command starts one or all of the following Network File System (NFS) server daemons. For more information about these daemon jobs, see i5/OS Network File System Support book, SC41-5714

SERVER(*ALL) should be used, which will start the daemons in the following order. (This order is the recommended order for starting the Network File System daemons.)

- The Remote Procedure Call (RPC) RPCBind daemon
- The block input/output (I/O) (BIO) daemon
- The generic security service (GSS) daemon
- The name registry (RGY) daemon
- The server (SVR) daemon
- The mount (MNT) daemon
- The network status monitor (NSM) daemon
- The network lock manager (NLM) daemon

If just one daemon is to be started, be sure the appropriate order for starting NFS daemons and the possible consequences of starting daemons in an order other than that specified above are understood. For more information about starting NFS daemons, see i5/OS Network File System Support book, SC41-5714

If the user attempts to start a daemon or daemons that are already running, they will not cause the command to fail, and it will continue to start other daemons that were requested to start. The command will issue diagnostic message CPDA1BA or CPDA1BD if the daemon is already running. However, for best results, end NFS daemons before attempting the STRNFSSVR command.

To determine if an NFS daemon is running, use the Work with Active Jobs (WRKACTJOB) command and look in the subsystem QSYSWRK for existence of the following jobs:

```
QNFSRPCD  The RPCBind daemon
QNFSBIOD  The block I/O (BIO) daemon
QNFSGSSD  The generic security service (GSS) daemon
QNFSRGYD  The name registry (RGY) daemon
QNFSNFSD  The NFS server (SVR) daemon
QNFSMNTD  The mount (MNT) daemon
QNFSNSMD  The network status monitor (NSM) daemon
QNFSNLMD  The network lock manager (NLM) daemon
```

Restrictions:

- The user must have input/output (I/O) system configuration (*IOSYSCFG) special authority to use this command.
- The user must be enrolled in the system distribution directory. Use the Add Directory Entry (ADDDIRE) command to enroll the user.

Top

Parameters

Keyword	Description	Choices	Notes
SERVER	Server daemon	*ALL, *RPC, *BIO, *GSS, *RGY, *SVR, *MNT, *NSM, *NLM	Required, Positional 1
NBRSVR	Number of server daemons	1-20, **1**	Optional, Positional 2
NBRBIO	Number of block I/O daemons	1-20, **1**	Optional
RTVRPCREG	Retrieve RPC registration	***NO**, *YES	Optional
STRJOBTIMO	Timeout for start of daemon	1-3600, **30**, *NOMAX	Optional

Top

Server daemon (SERVER)

Specifies the Network File System (NFS) daemon jobs to be started by this command. The specified daemon should not already be running.

***ALL** All NFS daemons will be started.

***RPC** The NFS RPCBind daemon will be started.

***BIO** Starts NFS block input/output (I/O) daemons. Additional daemons will be started if the number specified on the **Number of block I/O daemons (NBRBIO)** parameter is greater than the number of block I/O daemons already running on the system.

***GSS** The NFS generic security services daemon will be started.

***RGY** The NFS name resolution registry daemon will be started.

***SVR** Starts NFS server daemons. Additional daemons will be started if the number specified on the **Number of server daemons (NBRSVR)** parameter is greater than the number of server daemons already running on the system.

***MNT** The NFS mount daemon will be started.

***NSM** The NFS network status monitor daemon will be started.

***NLM** The NFS network lock manager daemon will be started.

This is a required parameter.

Top

Number of server daemons (NBRSVR)

Specifies the number of NFS server (*SVR) daemon jobs the user wants to have running. Additional daemons will be started if the number specified on this parameter is greater than the number of server daemons already running on the system. This parameter can only be used if SERVER(*SVR) or SERVER(*ALL) is specified.

1 One NFS server daemon job should be started if there are not already any NFS server daemons running.

1-20 Specify the number of NFS server daemon jobs the user wants to have running.

Top

Number of block I/O daemons (NBRBIO)

Specifies the number of NFS block input/output (I/O) (*BIO) daemon jobs the user wants to have running. Additional daemons will be started if the number specified on this parameter is greater than the number of block I/O daemons already running on the system. This parameter can only be used if SERVER(*BIO) or SERVER(*ALL) is specified.

1 One NFS block I/O daemon job should be started if there are not already any NFS block I/O daemons running.

1-20 Specify the number of NFS block I/O daemon jobs the user wants to have running.

Top

Retrieve RPC registration (RTVRPCREG)

Specifies whether to retrieve previously recorded registration information when the RPCBind daemon is started. If registration information is retrieved, any services already registered with the RPCBind daemon do not have to re-register with the RPCBind daemon. This parameter can only be used if SERVER(*RPC) or (SERVER(*ALL) is specified.

***NO** Do not retrieve registration information.

***YES** Retrieve registration information.

Top

Timeout for start of daemon (STRJOBTIMO)

Specifies the number of seconds to wait for each daemon to successfully start. If a daemon has not started within the timeout value, the command will fail.

30 Default seconds before timeout.

***NOMAX**
Wait forever for daemons to start; do not timeout.

1-3600 Specify a number of seconds to wait for daemons to start before timing out and failing the command. Timeout values less than 30 seconds are rounded up to 30 seconds.

Top

Examples

Example 1: Start All NFS Daemons

```
STRNFSSVR   SERVER(*ALL)  STRJOBTIMO(*NOMAX)
```

This command starts all NFS daemons, and waits forever for them to start. No daemons should be previously running.

Example 2: Start Only One Daemon

```
STRNFSSVR   SERVER(*MNT)
```

This command starts the NFS mount daemon, and waits up to the default of 30 seconds for it to start. The mount daemon should not be already running, and other daemons have been started in the appropriate order.

Top

Error messages

***ESCAPE Messages**

CPFA1B8
*IOSYSCFG authority required to use &1.

Top

Start Object Conversion (STROBJCVN)

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

Parameters
Examples
Error messages

The **Start Object Conversion (STROBJCVN)** command either converts user objects or checks which objects will require conversion. Objects are converted from the format used in a previous version, release, and modification level of the operating system to the format required for use in the current version, release, and modification level of the operating system.

The following types of objects in libraries are converted or checked by this command:

- Programs (original program model (OPM) and Integrated Language Environment (ILE))
- Service programs
- Modules

SQL data is converted or checked by this command. Programs, service programs, and SQL package (*SQLPKG) objects may contain SQL statements and data used for external stored procedures. Note that converting SQL data contained within a program object is handled separately from converting the program itself.

The following object types in directories are converted or checked by this command:

- Stream files with Java programs that were created to run on iSeries. Only stream files in the ″root″ (/), QOpenSys and user-defined file systems (UDFS) are included.

The user objects that are not converted with this command are automatically converted when they are first used.

Note: Using objects that are not yet converted will have one-time performance degradation while the conversion operation is performed.

Note: In most cases, the objects analyzed by Analyze Object Conversion (ANZOBJCVN) can be converted by the STROBJCVN command or on first touch. However, STROBJCVN will not convert the directories affected by new Unicode characters and casing rules in the target release since this conversion is automatically done by the operating system when necessary. Spooled files will not be converted by STROBJCVN, conversion is automatically done by the operating system when necessary.

Restrictions:

- Any primary or secondary independent ASP (ASPs 33-255) referenced by this command must be varied on and have a status of ′Available′ before running this command.
- Any user-defined file systems referenced by this command must be mounted before running this command.
- Objects located in a ′read-only′ user-defined file system will not be converted.
- You must have all object (*ALLOBJ) special authority to run this command.
- This command can be very long running. For this reason, it is suggested that this command be run in a batch job.

Top

Parameters

Keyword	Description	Choices	Notes
OPTION	Option	***CONVERT**, *CHECK	Optional, Positional 6
LIB	Library	*Name*, *ALLUSR, *NONE	Required, Positional 1
OBJTYPE	Object type	***ALL**, *FILE, *ALLPGM, *SQL	Optional, Positional 2
ASPDEV	ASP device	*Name*, *****, *ALLAVL, *CURASPGRP, *SYSBAS	Optional, Positional 3
OBJ	Object	*Path name*, ***NONE**	Optional, Positional 4
SUBTREE	Directory subtree	***ALL**, *NONE	Optional, Positional 5

Top

Option (OPTION)

Specifies whether conversion is performed or objects are checked to determine if conversion is required.

***CONVERT**
Objects specified on the **Library (LIB)** parameter and **Object (OBJ)** parameter are converted to the format required by the current version, release, and modification level of the operating system.

***CHECK**
Objects specified on the LIB parameter and OBJ parameter are checked to determine if they need to be converted. Messages are sent to the job log which includes the number of objects which have already been converted and the number of objects which still need to be converted.

Top

Library (LIB)

Specifies the user library for which objects are to be checked or converted to the format required for use in the current version, release, and modification level of the operating system.

***ALLUSR**
All user libraries are selected. All libraries with names that do not begin with the letter Q are selected 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 selected:

```
QDSNX       QRCLxxxxx   QUSRDIRDB   QUSRVI
QGPL        QSRVAGT     QUSRIJS     QUSRVxRxMx
QGPL38      QSYS2       QUSRINFSKR
QMGTC       QSYS2xxxxx  QUSRNOTES
QMGTC2      QS36F       QUSROND
QMPGDATA    QUSER38     QUSRPOSGS
```

```
QMQMDATA    QUSRADSM    QUSRPOSSA
QMQMPROC    QUSRBRM     QUSRPYMSVR
QPFRDATA    QUSRDIRCF   QUSRRDARS
QRCL        QUSRDIRCL   QUSRSYS
```

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.

***NONE**
No user library is selected. You can specify LIB(*NONE) if you only want to convert or check objects that are located in directories.

name Specifies the name of the user library whose objects are to be checked or converted.

Top

Object type (OBJTYPE)

Specifies which object types in the library should be converted.

If *CHECK is specified for the Option (OPTION) parameter, specification of this parameter is ignored.

If *NONE is specified for the Library (LIB) parameter, specification of this parameter is ignored.

***ALL** All program (*PGM) objects, service program (*SRVPGM) objects, module (*MODULE) objects, and database file (*FILE) objects in the specified library are converted. In addition, the stored SQL information in all *PGM, *SRVPGM, and *SQLPKG objects that contain SQL statements, as well as *PGM and *SRVPGM objects that are used to implement external stored procedures, is converted.

***FILE** Only database file member objects in the specified library are converted.

***ALLPGM**
All program (*PGM) and service program (*SRVPGM) objects in the specified library are converted.

***SQL** The stored SQL information in all *PGM, *SRVPGM, and *SQLPKG objects that contain SQL statements, as well as *PGM and *SRVPGM objects that are used to implement external stored procedures, is converted.

Top

ASP device (ASPDEV)

Specifies the auxiliary storage pool (ASP) device where storage is allocated for the library to be checked or converted. If the library is in an ASP that is not part of the thread’s library name space, this parameter must be specified to ensure the correct library is checked or converted.

Note: This parameter does not apply for the objects specified in the OBJ parameter since the independent ASP name is part of the path name of the object.

***** The ASPs that are currently part of the thread’s library name space will be searched to find the library. 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.

***ALLAVL**
All available ASPs will be searched. This includes the system ASP (ASP 1), all defined basic user ASPs (ASPs 2-32), and all available primary and secondary ASPs (ASPs 33-255 with a status of ’Available’).

***CURASPGRP**
If the thread has an ASP group, the primary and secondary ASPs in the thread’s ASP group will be searched to find the library. The system ASP (ASP 1) and defined basic user ASPs (ASPs 2-32) will not be searched. If no ASP group is associated with the thread an error will be issued.

***SYSBAS**
The system ASP (ASP 1) and all defined basic user ASPs (ASPs 2-32) will be searched to find the library. No primary or secondary ASPs will be searched, even if the thread has an ASP group.

name

Specify the name of the primary or secondary ASP device to be searched. The primary or secondary ASP must have been activated (by varying on the ASP device) and have a status of ’Available’. The system ASP (ASP 1) and defined basic user ASPs (ASPs 2-32) will not be searched.

Top

Object (OBJ)

Specifies the path name of the object or a pattern to match the name of the objects to be checked or converted.

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 topic collection in the Programming category in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/.

Note: If the last component in the path name is a symbolic link, the symbolic link object is checked or converted, but the object pointed to by the symbolic link is not checked or converted.

***NONE**
No object is selected. You can specify OBJ(*NONE) if you only want to convert or check objects that are located in libraries.

path-name
Specify the path name of the object to be checked or converted.

Top

Directory subtree (SUBTREE)

Specifies whether or not to check or convert the objects within the subtree if the object specified by the **Object (OBJ)** parameter is a directory.

***ALL** The objects specified by OBJ will be checked or converted, if appropriate. If the object is a directory, its contents as well as the contents of all of its subdirectories will be checked or converted.

Once the command has begun processing a specific directory subtree, the objects which will be found and processed may be affected by operations that update the organization of objects within the specified directory tree. This includes, but is not limited to, the following:

- Adding, removing, or renaming object links
- Mounting or unmounting file systems
- Updating the effective root directory for the process calling the command
- Updating the contents of a symbolic link

In order to process the directory subtree, the system code may increase the process-scoped maximum number of file descriptors that can be opened during processing. This is done so that the command is not likely to fail due to a lack of descriptors. This process-scoped maximum value is not reset when the command completes.

***NONE**
The objects specified by OBJ will be checked or converted, if appropriate. If the object is a directory, it is checked or converted if appropriate, but its contents are not checked or converted.

Top

Examples

Example 1: Start Conversion of All Objects in One Library

```
STROBJCVN   OPTION(*CONVERT)  LIB(LIB1)  OBJ(*NONE)
```

This command converts all original program model (OPM) and integrated language environment (ILE) programs, service programs and modules contained in the LIB1 user library to the format required by the current release of the operating system.

In addition, the stored SQL information in all *PGM, *SRVPGM, and *SQLPKG objects that contain SQL statements, as well as *PGM and *SRVPGM objects that are used to implement external stored procedures, is converted to the format required by the current release of the operating system.

No objects in directories will be converted.

Example 2: Start Conversion of SQL Objects

```
STROBJCVN   OPTION(*CONVERT)  LIB(LIB2)  OBJTYPE(*SQL)
```

This command converts the stored SQL information in all *PGM, *SRVPGM, and *SQLPKG objects in library LIB2 that contain SQL statements, as well as *PGM and *SRVPGM objects in library LIB2 that are used to implement external stored procedures, to the format required by the current release of the operating system.

Example 3: Start Conversion of All Programs

```
STROBJCVN   OPTION(*CONVERT)  LIB(*ALLUSR)  OBJTYPE(*ALLPGM)
            ASPDEV(*ALLAVL)  OBJ('/')  SUBTREE(*ALL)
```

This command converts all OPM and ILE program (*PGM) and service program (*SRVPGM) objects in all user libraries on all available auxiliary storage pools, include the system ASP, all defined basic user ASPs, and all primary and secondary independent ASPs.

In addition, all Java programs in the ″root″ (/) file system ('/' directory, its contents and the contents of all its subdirectories) will be converted.

Depending on the number of user libraries and the number of directories on the system, this command could take a very long time to run.

Example 4: Check How Many Objects in a Library Need to be Converted

```
STROBJCVN   OPTION(*CHECK)  LIB(LIB1)  OBJ(*NONE)
```

This command checks all OPM and ILE programs, service programs, and modules in library LIB1. It will send a completion message that includes the total number of objects that were checked, the number of objects that have already been converted, and the number of objects that still need to be converted to the format required by the current release of the operating system.

Example 5: Start Conversion in a Specific Directory

```
STROBJCVN   OPTION(*CONVERT)  LIB(*NONE)  OBJ('/MYDIR')  SUBTREE(*ALL)
```

This command converts all Java programs in the /MYDIR directory, its contents, as well as the contents of all of its subdirectories will be analyzed.

Top

Error messages

***ESCAPE Messages**

CPF218C
&1 not a primary or secondary ASP.

CPF9833
*CURASPGRP or *ASPGRPPRI specified and thread has no ASP group.

CPFB0E1
Not all objects were checked.

CPFB0E2
Not all objects were converted.

Top

Start Pass-Through (STRPASTHR)

Where allowed to run: Interactive environments (*INTERACT *IPGM *IREXX *EXEC)
Threadsafe: No

Parameters
Examples
Error messages

The Start Pass-Through (STRPASTHR) command allows you to pass through to a target system where you can sign on as if you were attached locally. For information on configuring or operating the pass-through function, refer to the Remote Work Station Support book.

Restriction:

1. This command cannot be entered at a work station with a display that has 12 lines by 80 characters.

Top

Parameters

Keyword	Description	Choices	Notes
RMTLOCNAME	Remote location	*Communications name*, *CNNDEV	Required, Positional 1
CNNDEV	APPC device	Single values: ***LOC** Other values (up to 16 repetitions): *Name*	Optional
VRTCTL	Virtual controller	*Name*, ***NONE**	Optional, Positional 2
VRTDEV	Virtual display device	Single values: ***NONE** Other values (up to 32 repetitions): *Name*	Optional, Positional 3
MODE	Mode	*Communications name*, ***NETATR**	Optional
LCLLOCNAME	Local location	*Communications name*, ***LOC**, *NETATR	Optional
RMTNETID	Remote network identifier	*Communications name*, ***LOC**, *NETATR, *NONE	Optional
SRQ10PGM	System request program	Single values: ***SRQMNU** Other values: *Qualified object name*	Optional
	Qualifier 1: System request program	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
RMTUSER	User profile	*Character value*, ***NONE**, *CURRENT	Optional
RMTPWD	User password	*Character value*, ***NONE**	Optional
RMTINLPGM	Initial program to call	*Name*, ***RMTUSRPRF**, *NONE	Optional
RMTINLMNU	Initial menu	*Name*, ***RMTUSRPRF**, *SIGNOFF	Optional
RMTCURLIB	Current library	*Name*, ***RMTUSRPRF**	Optional
PASTHRSCN	Display option	*Character value*, ***YES**, *NO	Optional

Top

Remote location (RMTLOCNAME)

Specifies one of the following:

- If (*LOC) is specified for the **APPC device** prompt (CNNDEV parameter), the RMTLOCNAME parameter specifies the name of the remote location that is the target of the pass-through session.

- If any devices are specified on the CNNDEV parameter, the RMTLOCNAME parameter specifies the first system to do intermediate pass-through routing.

remote-location-name
Specify the remote location name that is the target of the pass-through session or the first system that does intermediate pass-through routing. APPN determines the route to this location.

***CNNDEV**
Use the APPC devices specified with the **APPC device** prompt (CNNDEV parameter).

Top

APPC device (CNNDEV)

Specifies the names of the device descriptions that connect the first system to do pass-through routing with the target system. If *CNNDEV is specified on the Remote location prompt (RMTLOCNAME parameter), the first device specified on this parameter is on the source system. If the name of a remote location is specified on the Remote location prompt (RMTLOCNAME parameter), the first device specified on this parameter is on the system in that remote location. If another system connects the source system to the target system and pass-through must establish the intermediate sessions, then you must specify a list of APPC device descriptions. The APPC device names must be listed in the order that the systems are passed through to get to the target system.

You can enter multiple values for this parameter.

***LOC** The **Remote location** prompt (RMTLOCNAME parameter) specifies the name of the remote location that is the target of the pass-through session.

device-name(s)
Specify the names of the device descriptions that complete the route from the source system to the target system. Up to 16 names can be specified.

Top

Virtual controller (VRTCTL)

Specifies the name of the virtual controller on the remote system that is used to do pass-through jobs. If you specify a virtual controller, one of the virtual display devices attached to it is selected for the pass-through job. A device on the target system is selected based on a comparison of device type and model.

This entry is mutually exclusive with an entry on the **Virtual display device** prompt (VRTDEV parameter); *NONE must be specified when one or more devices are specified on the **Virtual display device** prompt (VRTDEV parameter).

***NONE**
No controller is specified. If *NONE is also specified on the **Virtual display device** prompt (VRTDEV parameter), you are requesting automatic configuration.

virtual-controller-name
Specify the name of the virtual controller description on the target system.

Top

Virtual display device (VRTDEV)

Specifies one or more devices on the target system that are connected to a virtual controller used for the pass-through session. A device on the target system from the list is selected based on a comparison of device type and model.

This entry is mutually exclusive with an entry on the **Virtual controller** prompt (VRTCTL parameter); *NONE must be specified when a controller name is specified on the **Virtual controller** prompt (VRTCTL parameter).

You can enter multiple values for this parameter.

***NONE**
No device names are specified. If *NONE is also specified on the **Virtual controller** prompt (VRTCTL parameter), you are requesting automatic configuration.

virtual-display-device-name
Specify the names of the virtual display device descriptions on the target system. Up to 32 names can be specified.

Top

Mode (MODE)

Specifies the mode name used.

***NETATR**
The mode in the network attributes is used.

mode-name
Specify a mode name.

Top

Local location (LCLLOCNAME)

Specifies the local location name.

***LOC** The local location name is chosen by the system.

***NETATR**
The LCLLOCNAME value specified in the system network attributes is used.

APPN-location-name
Specify the local location name that is associated with the source system. If the local location name is not valid, an escape message is sent.

Top

Remote network identifier (RMTNETID)

Specifies the network ID of the network where the remote location resides.

***LOC** Any remote network ID for the remote location may be used.

***NETATR**
The LCLNETID value specified in the system network attributes is used.

***NONE**
The remote location does not support network identifiers.

APPN-network-identifier
Specify a remote network ID.

Top

System request program (SRQ10PGM)

Specifies that the system request menu is displayed, or specifies the name of the program that starts on the source system when system request option 10 is selected.

The user program can display a menu that allows you to select the system you want to access, and then transfer to a group job that sends the Start Pass-Through (STRPASTHR) command to the desired system. For more information, see the Remote Work Station Support book.

***SRQMNU**
The system request menu on the source system is displayed.

program-name
Specify the program and library names of the program started when system request option 10 is selected.

The possible library values are:

***LIBL** The library list is used to locate the program.

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

library-name
Specify the library where the program is located.

Top

User profile (RMTUSER)

Specifies the user profile for automatic sign-on to the target system. If a profile is specified for this parameter and password security is active on the target system, (*NONE) is not valid for the User password prompt (RMTPWD parameter).

***NONE**
No user profile name is sent, and no automatic sign-on occurs.

***CURRENT**
The user profile of the job using this command is sent. If the target system allows it, and the user profile exists on the target system, and the password specified in the RMTPWD parameter is valid for the profile, the user is automatically signed on. Otherwise, the user is presented with a sign-on display on the target system or a failure message on the source system, depending on the configuration of the target system.

profile-name
Specify a user profile name to use that exists on the target system. If the target system allows it, and if the user profile exists on the target system, the user is automatically signed on. Otherwise, the user is presented with a sign-on display on the target system or a failure message on the

source system, depending on the configuration of the target system. If a profile is specified and password security is active on the target system, a password must be specified, even if the profile specified is the same as the current profile.

Top

User password (RMTPWD)

Specifies the password being sent to the target system.

***NONE**
The system does not send a password. If a profile is specified on the **User profile** prompt (RMTUSER parameter), and password security is active on the target system, this value is not allowed.

password
Specify a password being sent to the target system to verify the sign-on of the user specified in the RMTUSER parameter. This password is encrypted before being sent across the communication line.

Top

Initial program to call (RMTINLPGM)

Specifies the program that is called immediately after you sign on.

***RMTUSRPRF**
The initial program specified in the remote user profile is called immediately after automatic sign-on.

***NONE**
No program is run before the initial menu is shown, even if an initial program is specified in the remote user profile.

program
Specify the name of a program that is called immediately after automatic sign-on.

Top

Initial menu (RMTINLMNU)

Specifies the first menu shown when you are automatically signed on to the target system after the initial program is run.

***RMTUSRPRF**
The initial menu specified in the remote user profile is shown immediately after the initial program is run.

***SIGNOFF**
A menu is not shown after the initial program is run, even if an initial menu is specified in the remote user profile. After the initial program ends, the user is signed off, and the pass-through session ends.

menu Specify the menu that is shown immediately after the initial program is run.

Top

Current library (RMTCURLIB)

Specifies the name of the library that becomes the current library in the library list of the job after automatic sign-on.

***RMTUSRPRF**
The current library specified in the remote user profile becomes the current library in the library list after automatic sign-on.

library
Specify the library that becomes the current library in the library list after automatic sign-on.

Top

Display option (PASTHRSCN)

Specifies whether the pass-through display and associated status messages appear before the pass-through session is established.

***YES** The pass-through display and information messages are shown before the pass-through session is established.

***NO** The pass-through display and information messages are not shown before the pass-through session is established.

Top

Examples

Example 1: Pass-Through to Toronto

```
STRPASTHR   RMTLOCNAME(*CNNDEV)  CNNDEV(DET CHI TOR)  VRTCTL(VWSC)
```

This command specifies starting a pass-through to the Toronto system by going through Detroit and Chicago. More information is in the **Remote Work Station Support** book located in the Information Center.

Example 2: Pass-Through to Detroit

```
STRPASTHR   RMTLOCNAME(DETROIT)  VRTCTL(VWSC)
```

This command specifies a pass-through to the Detroit system. APPN establishes the route to Detroit.

Example 3: Pass-Through to Toronto

```
STRPASTHR   RMTLOCNAME(DETROIT)  CNNDEV(CHI TOR)  VRTCTL(VWSC)
```

This command specifies another way to pass-through to the Toronto system by going through Chicago and Detroit. APPN establishes the route to Detroit.

Example 4: Pass-Through to Detroit

```
STRPASTHR   RMTLOCNAME(DETROIT)  RMTUSER(*CURRENT)
```

This command specifies a pass-through to the DETROIT system and an automatic sign-on using the user profile with the same name as the one currently used on the source system. It also specifies that the DETROIT system automatically configures a virtual device for the pass-through session, since a virtual controller or virtual device was not specified.

Top

Error messages

***ESCAPE Messages**

CPF2702
Device description &1 not found.

CPF2703
Controller description &1 not found.

CPF5383
Mode &7 specified for device &4 not valid.

CPF5536
System cannot automatically select an APPC device description for the remote location.

CPF5546
Class-of-service for device &4 not valid.

CPF8901
Virtual device &1 not varied on.

CPF8902
Virtual device &1 not available.

CPF8903
Device &1 not valid for pass-through.

CPF8904
Pass-through request not accepted.

CPF8905
Pass-through not allowed on this system.

CPF8906
Error during session initialization. Reason code &1.

CPF8907
Communications failure for device &1.

CPF8908
Controller &1 not varied on.

CPF8909
Old software release. Pass-through ended.

CPF8910
Controller &1 not valid for pass-through.

CPF8911
Communications failure. Session was not started.

CPF8912
Pass-through session ended. Reason code &1.

CPF8913
Pass-through ended abnormally.

CPF8916
Cannot select virtual device &1 at system &2.

CPF8917
Not authorized to &1.

CPF8918
Job canceled at system &1.

CPF8919
Device &1 not accessed by system &2.

CPF8920
Pass-through failed. &1 must be varied off and on.

CPF8921
APPC failure. Failure code is &3.

CPF8922
Negative response from device &1 at system &2.

CPF8923
Data stream received at system &1 not valid.

CPF8924
No available virtual controller.

CPF8925
Device &1 not created.

CPF8928
Device &1 could not be changed.

CPF8929
Device &1 could not be varied on.

CPF8931
Location &1 not an APPC location.

CPF8932
Device &1 must be non-networking APPC device.

CPF8933
Route to specified location not found.

CPF8935
Pass-through not allowed to system &1.

CPF8936
Pass-through failed for security reasons.

CPF8937
Automatic sign on not allowed.

CPF8938
Error in QRMTSIGN program. Pass-through failed.

CPF8939
Trying to send too much data.

CPF8940
Cannot automatically select virtual device.

CPF8941
Incorrect internal use of pass-through.

CPF8943
Pass-through not allowed from server TELNET session.

*STATUS Messages

CPI8901
No matching device on remote system. Function limited.

CPI8902
Pass-through started at system &1.

CPI8903
Virtual device &1 selected at system &2.

CPI8906
Automatic sign-on not allowed.

Top

Start PC Command (STRPCCMD)

Where allowed to run:

- Interactive job (*INTERACT)
- Interactive program (*IPGM)
- Using QCMDEXEC, QCAEXEC, or QCAPCMD API (*EXEC)

Threadsafe: No

Parameters
Examples
Error messages

The Start PC Command (STRPCCMD) command allows you to run a single application, a DOS command, or an OS/2 command on an attached personal computer.

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 STRPCCMD

None

Top

Parameters

Keyword	Description	Choices	Notes
PCCMD	PC command	*Character value*	Required, Positional 1
PAUSE	Pause	***YES**, *NO	Optional

Top

PC command (PCCMD)

Specifies the valid DOS command, OS/2 command, or personal computer application you want to run. If you are using the STRPCCMD command on a command line and the command or application name contains special characters, you must enclose the command in apostrophes. Special characters include blanks, commas, and colons.

If you select the option to start a PC command from the System i Access Organizer menu, you do not need to enclose the command in apostrophes unless it ends in a colon. The ending colon is ignored unless the entire PC command is enclosed in apostrophes.

This is a required parameter.

Top

Pause (PAUSE)

Specifies whether the computer should pause after running a command.

The possible values are:

***YES** The computer will pause after running the command before returning to the System i Access Organizer menu.

***NO** The computer returns immediately to the System i Access Organizer menu after the command runs.

Top

Examples

None

Top

Error messages

None

Top

Start Organizer (STRPCO)

Where allowed to run:

- Interactive job (*INTERACT)
- Interactive program (*IPGM)
- Using QCMDEXEC, QCAEXEC, or QCAPCMD API (*EXEC)

Threadsafe: No

Parameters
Examples
Error messages

The Start System i Access Organizer (STRPCO) command starts the System i Access Organizer on the host system.

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 STRPCO

None

Top

Parameters

Keyword	**Description**	**Choices**	**Notes**
PCTA	Text Assist	*Character value*, ***YES**, *NO	Optional, Positional 1

Top

Text Assist (PCTA)

Specifies whether you are going to use the Personal Computer Text Assist.

The possible values are:

***YES** The Personal Computer Text Assist is going to be used.

***NO** The Personal Computer Text Assist is not going to be used.

Top

Examples

None

Top

Error messages

None

Top

Start Performance Explorer (STRPEX)

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

Parameters
Examples
Error messages

The Start Performance Explorer (STRPEX) command starts a new Performance Explorer session or resumes a suspended Performance Explorer session.

Restrictions:

1. This command is shipped with public *EXCLUDE authority.
2. 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 (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.
3. The following user profiles have private authorities to use the command:
   - QPGMR
   - QSRV
4. Two threads within the same job will not be allowed to run STRPEX at the same time. The thread that issued STRPEX first will run the command to completion while the second STRPEX waits.

Top

Parameters

Keyword	Description	Choices	Notes
SSNID	Session ID	*Name*	Required, Key, Positional 1
OPTION	Option	***NEW**, *INZONLY, *RESUME	Optional, Positional 2
DFN	Definition	*Name*, ***SELECT**	Optional
FTR	Filter	*Name*, ***NONE**, *SELECT	Optional

Top

Session ID (SSNID)

Specifies an identifier for this Performance Explorer session. This name must be unique within the active sessions of the Performance Explorer tool.

This is a required parameter.

session-identifier
Specify the session identifier for a new or suspended Performance Explorer session.

Top

Option (OPTION)

Specifies whether a new session should be started or a session which is currently suspended should be resumed.

This is a required parameter.

***NEW** A new session of the Performance Explorer tool is started. Performance data collection begins as soon as the session has been established.

***INZONLY**
A new Performance Explorer session is started, but once the session is established, it is suspended. This option allows the user to perform the setup for a particular session before the scenario to be monitored is started. To begin data collection, the user must invoke this command again specifying the same session identifier and OPTION(*RESUME).

***RESUME**
A suspended Performance Explorer session is resumed. The session was suspended either by specifying OPTION(*SUSPEND) on the ENDPEX command or by specifying OPTION(*INZONLY) on a previous STRPEX command.

Top

Definition (DFN)

Specifies the name of the Performance Explorer definition to use when starting a new Performance Explorer session. The definition identifies the performance data to be collected. This parameter is required in order to start a new Performance Explorer session. This parameter cannot be specified when resuming a suspended session, since the suspended session already has an associated Performance Explorer definition.

name Specify the name of the Performance Explorer definition.

Top

Filter (FTR)

Specifies the name of the Performance Explorer filter to use when starting a new Performance Explorer session. The filter determines which events are collected based on the filter values which are compared to the event's actual data. If a filter is not specified, then all events in the definition are collected. This parameter cannot be specified when resuming a suspended session.

***SELECT**
A list of existing filters will be displayed for user selection. This value is valid only if the command is run in an interactive job.

name Specify the name of the Performance Explorer filter.

Top

Examples

Example 1: Start a New Session

```
STRPEX   SSNID(TESTRUN2)  DFN(NEWDESC)  OPTION(*NEW)
         FTR(MYFILTER)
```

This command starts a new session of the performance explorer using the criteria identified in a definition named NEWDESC and a filter named MYFILTER. The new session name is TESTRUN2.

Example 2: Resume a Suspended Session

```
STRPEX   SSNID(TESTRUN1)  OPTION(*RESUME)
```

This command resumes an already existing session of the performance explorer named TESTRUN1.

Top

Error messages

Error messages for RMVPEXDFN

***ESCAPE Messages**

CPFAF05
STRPEX command was not successful. Reason code is &1. See details for more information.

CPFAF11
Unable to locate program, object, library, or file member as specified in the definition or filter.

Top

Start Performance Collection (STRPFRCOL)

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

Parameters
Examples
Error messages

The Start Performance Collection (STRPFRCOL) command starts the system-level collection of performance data by Collection Services. The properties of the system-level collection are controlled by the Collection Services configuration, which can be changed using the Configure Performance Collection (CFGPFRCOL) command. The data included in the system-level collection is determined by the value specified on the Collection profile parameter.

Performance data collection is conducted by the Collection Services server job (QYPSPFRCOL). If this job is not active, it will be started as a result of this command. If the job is active, any change to the collection profile will take effect immediately and the collection will continue uninterrupted.

QYPSPFRCOL creates a management collection object (*MGTCOL) to store performance data. Data collected will include both the system-level collection profile categories as well as data collected on behalf of client applications (for example, IBM System i Navigator monitors or Performance Collector APIs). All data is stored in the current management collection object. This data may be processed using the Create Performance Data (CRTPFRDTA) command to generate the performance database files.

Cycling the performance collection will cause the Collection Services server job to create a new management collection object prior to implementing the specified Collection profile. Note: cycling may result in the loss of one interval of data and will also impact any client applications that are using Collection Services.

Top

Parameters

Keyword	Description	Choices	Notes
COLPRF	Collection profile	***CFG**, *MINIMUM, *STANDARD, *STANDARDP, *ENHCPCPLN, *CUSTOM	Optional, Positional 1
CYCCOL	Cycle collection	***NO**, *YES	Optional, Positional 2

Top

Default collection profile (COLPRF)

Specifies the collection profile. This will determine which categories will be included in the system-level collection. Specifying a profile for this parameter will change the default collection profile in the Collection Services configuration. The default collection profile may also be changed using the Configure Performance Collection (CFGPFRCOL) command.

***CFG**
The collection will use the currently configured value. If the configured value has never been modified, the collection profile will be *STANDARDP.

***MINIMUM**
The minimum data collection recommended. Includes the following categories: *SYSBUS, *POOL, *HDWCFG, *SYSCPU, *SYSLVL, *JOBMI, *JOBOS, *DISK, and *IOPBASE.

***STANDARD**
The standard profile includes all categories which are typically needed by the Performance Tools for System i5, with the exception of communications data. It includes all categories in the *MINIMUM profile, as well as the following categories: *POOLTUNE, *SUBSYSTEM, *SNADS, *LCLRSP, *APPN, *SNA, *TCPBASE, *USRTNS, and *LPAR. The category *DOMINO will be included if the product Domino for System i5 has been installed on the system. The category *HTTP will be included if the product IBM HTTP Server for System i5 (powered by Apache) has been installed on the system.

***STANDARDP**
The standard plus profile includes all categories which are typically needed by the Performance Tools for System i5, including communications data. It includes all categories in the *STANDARD profile, as well as the following categories: *IPCS, *CMNBASE, *CMNSTN, *CMNSAP, *TCPIFC, and *DPS.

***ENHCPCPLN**
The enhanced capacity planner profile includes all categories in the *STANDARDP profile, with the addition of *INTPEX.

***CUSTOM**
The custom profile includes categories as defined by the user. This profile must be defined using the Collection Services function in IBM System i Navigator.

Top

Cycle collection (CYCCOL)

Specifies whether the collection should be cycled. Cycling the collection will cause data to be collected in a new management collection object (*MGTCOL).

***NO** Data will continue to be collected in the current collection object.

***YES** Data will be collected in a new collection object.

Top

Examples

Example 1: Starting the Performance Collection

```
STRPFRCOL
```

This command will cause the Collection Services server job (QYPSPFRCOL) to start the system-level collection of performance data using the currently configured default collection profile.

Example 2: Starting with Collection Profile and Cycle Collection

```
STRPFRCOL COLPRF(*MINIMUM) CYCCOL(*YES)
```

This command will cause the Collection Services server job (QYPSPFRCOL) to cycle and begin collecting performance data in a new management collection object (*MGTCOL) using the *MINIMUM collection profile. Note: this command will also change the configured default collection profile to *MINIMUM.

Top

Error messages

***ESCAPE Messages**

CPF3CF2
Error(s) occurred during running of &1 API.

CPFB94A
Collector communications error. Reason code &1.

Top

Start Performance Trace (STRPFRTRC)

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

Parameters
Examples
Error messages

The Start Performance Trace (STRPFRTRC) command is a simplified interface to the TRCINT command for collecting Multiprogramming level (MPL) and Transaction trace data. This command collects the same performance trace data as was collected in previous releases by the Start Performance Monitor (STRPFRMON) command.

The trace started by this command creates and uses trace table QPM_STRPFRTRC.If the trace table exists, any existing data will be deleted before this trace begins.

The trace can be stopped and the data can be written to a data base file by using the End Performance Trace (ENDPFRTRC) command.

Restrictions:

- This command is shipped with public *EXCLUDE authority.
- The following user profiles have private authorities to use the command:
  – QSRV

Top

Parameters

Keyword	Description	Choices	Notes
SIZE	Trace table size	Single values: ***CALC**, *MAX Other values: *Element list*	Optional, Positional 1
	Element 1: Number of units	1-998000	
	Element 2: Unit of measure	***KB**, *MB	
OMTTRCPNT	Omit trace points	***NONE**, *RSCMGT	Optional
JOBTYPE	Job types	Single values: *NONE, *ALL Other values (up to 12 repetitions): ***DFT**, *ASJ, *BCH, *EVK, *INT, *MRT, *RDR, *SBS, *SYS, *WTR, *PDJ, *PJ, *BCI	Optional
JOBTRCITV	Job trace interval	0.1-9.9, **0.5**	Optional

Top

Trace table size (SIZE)

Specifies the size of the trace table.

Note: The storage indicated on this parameter is immediately allocated from the system auxiliary storage pool (ASP 1). Refer to the Trace Internal (TRCINT) command SIZE parameter for additional information regarding the setting of trace table size.

Specifying a size of less than 16 megabytes is not recommended.

Single values

***CALC**
The minimum trace table size is determined based on the processor group of your system.

***MAX** The trace table is set to the maximum size of 258048 megabytes.

Element 1: Number of units

Specify the size of the trace table.

1-998000
Specify the size of the trace table in kilobytes or megabytes.

Element 2: Unit of measure

Specify whether the value specified for the first element of this parameter should be treated as number of kilobytes or number of megabytes.

***KB** The trace table size is specified in kilobytes. The valid range is 128 through 998000.

***MB** The trace table size is specified in megabytes. The valid range is 1 through 258048.

Top

Omit trace points (OMTTRCPNT)

Specifies trace points whose trace records are to be excluded.

***NONE**
No trace points are to be excluded.

***RSCMGT**
Resource management trace points (seize/lock conflict data) will be excluded from the trace.

Top

Job types (JOBTYPE)

Specifies the types of jobs for which trace data is to be collected for use in the batch job trace report.

Note: The value *DFT includes the values *ASJ, *BCH, *EVK, *MRT, *PDJ, *PJ and *BCI. The value *BCH includes the values *EVK, *MRT, *PDJ, *PJ, and *BCI.

Single values

***NONE**
No jobs are to be traced.

***ALL** All job types are to be traced.

Other values (up to 12 repetitions)

***DFT** Batch and autostart jobs are traced.

***ASJ** Autostart jobs are traced.

***BCH** Batch jobs are traced.

***EVK** Jobs started by a procedure start request are traced.

***INT** Interactive jobs are traced.

***MRT** Multiple requester terminal jobs are traced.

***RDR** Reader jobs are traced.

***SBS** Subsystem monitor jobs are traced.

***SYS** System jobs are traced.

***WRT** Writer jobs are traced.

***PDJ** Print driver jobs are traced.

***PJ** Prestart jobs are traced.

***BCI** Batch immediate jobs are traced.

Top

Job trace interval (JOBTRCITV)

Specifies the time interval (in CPU seconds) to be used between each collection of the job trace data.

0.5 A time slice interval value of 0.5 CPU seconds is used.

0.1-9.9 Specify the trace interval to be used, in CPU seconds.

Top

Examples

```
STRPFRTRC   SIZE(*CALC)
```

This command starts the collection of performance trace data. The trace table size may be adjusted to the calculated minimum. This example will result in the same trace table size and data as would STRPFRMON TRACE(*ALL) DMPTRC(*NO).

Top

Error messages

***ESCAPE Messages**

CPF0A2A
Performance trace already started

Also refer to the TRCINT command for other messages.

Top

Start Program Export List (STRPGMEXP)

Parameters
Examples
Error messages

The Start Program Export List (STRPGMEXP) binder definition statement starts a list of exports in a service program export block.

Top

Parameters

Keyword	Description	Choices	Notes
PGMLVL	Program level	***CURRENT**, *PRV	Optional, Positional 1
LVLCHK	Signature level check	***YES**, *NO	Optional, Positional 2
SIGNATURE	Signature	*Character value*, ***GEN**	Optional, Positional 3

Top

Program level (PGMLVL)

Specifies the service program level for this export block.

***CURRENT**
This service program export block contains exports for the current service program.

***PRV** This service program export block contains exports for a previous version of the service program.

Top

Signature level check (LVLCHK)

Specifies whether a level check is performed on the export block.

***YES** The service program export block is level checked by generating a nonzero signature.

***NO** The service program export block is not level checked. A zero signature is generated.

Top

Signature (SIGNATURE)

Specifies the signature for the export block.

***GEN** If *YES is specified for the **Signature level check (LVLCHK)** parameter, the system generates a nonzero signature for the export block. If *NO is specified for the LVLCHK parameter, this value is required and the system generates a zero signature for the export block.

hexadecimal-character-value

The signature value for the export block is set to the specified string of hexadecimal digits and is not generated by the system. If the specified value is less than 32 hexadecimal digits in length, the system pads it on the left with hexadecimal zeros to 32 digits. If the specified value is greater than 32 hexadecimal digits in length, the system truncates it on the right to 32 digits.

character-value

The signature value for the export block is set to the EBCDIC character codes of the given signature-value and is not generated by the system. If the signature-value is less than 16 characters in length, the system pads the signature-value on the right with spaces to 16 characters. If the signature-value is greater than 16 characters in length, the system truncates it on the right to 16 characters.

Top

Examples

```
STRPGMEXP   PGMLVL(*CURRENT)  LVLCHK(*YES)  SIGNATURE(*GEN)
```

This binder definition statement marks the beginning of a list of exported variables or procedures for a service program. This service program export block contains exports for the current service program. Level checking will be performed using a signature generated by the operating system.

Top

Error messages

None

Top

Start Programmer Menu (STRPGMMNU)

Where allowed to run: Interactive environments (*INTERACT *IPGM *IREXX *EXEC)
Threadsafe: No

Parameters
Examples
Error messages

The Start Programmer Menu (STRPGMMNU) command shows the programmer menu. This command can be used instead of the CALL QPGMMENU function, and allows you to pass parameters to specify and control the data which appears in the associated fields on the programmer menu.

NOTES:

1. A user exit program can be called instead of submitting a job when option 3 is selected.
2. The first four parameters control the defaults that appear when the menu is first displayed.

More information about using the Programmer Menu is in the CL topic collection in the Programming category in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/.

Top

Parameters

Keyword	Description	Choices	Notes
SRCFILE	Source file	*Name*, ***DFT***	Optional, Positional 1
SRCLIB	Source library	*Name*, ***LIBL***, *CURLIB	Optional, Positional 2
OBJLIB	Object library	*Name*, ***DFT***, *CURLIB	Optional, Positional 3
JOBD	Job description	*Name*, ***USRPRF***	Optional, Positional 4
ALWUSRCHG	Allow changes	***YES***, *NO	Optional
EXITPGM	Option 3 exit program	Single values: ***NONE*** Other values: *Qualified object name*	Optional
	Qualifier 1: Option 3 exit program	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL***, *CURLIB	
DLTOPT	Delete option	***DLT***, *PROMPT, *NODLT	Optional

Top

Source file (SRCFILE)

Specifies an existing source file that contains source file members to be updated or to which new source file members are to be added.

DFT This is the default for the type being specified on the menu. This field is blank when shown on the display station.

name Specify the name of the source file to be updated.

Top

Source library (SRCLIB)

Specifies the library that is searched for the source file.

***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 entry exists in the library list, QGPL is used.

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

Top

Object library (OBJLIB)

Specifies the library that is to contain the object.

***DFT** Blanks appear for this field. The library used depends on the menu option you selected.

***CURLIB**
The current library for the job is the library that is to contain the object.

name Specify the name of the library that is to contain the created object.

Top

Job description (JOBD)

Specifies the job description used with the job being submitted.

***USRPRF**
The job description defined in the user profile of the user running the STRPGMMNU command is used for the job.

name Specify the name of the job description used for the job. The job description is found through the library list being used by the job.

Top

Allow changes (ALWUSRCHG)

Specifies whether the menu display fields you specified on the previous parameters in this command can be changed by the user.

***YES** Values on the display can be changed.

***NO** The display fields cannot be changed.

Top

Option 3 exit program (EXITPGM)

Specifies the user-written program that is called as an exit program in place of submitting a batch job when menu option 3 is selected. When the exit program is called, it receives parameters that are sent by the programmer menu. More information about the EXITPGM parameter is in the CL topic collection in the Programming category in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/.

Single values

***NONE**
No user-written program is called; a batch job is submitted. When *NONE is specified, *DLT must be specified for the **Delete option (DLTOPT)** parameter.

Qualifier 1: Option 3 exit program

name Specify the name of the program called when option 3 is selected, instead of submitting the create command as a batch job. When a value is specified on this parameter, the text that appears on the menu for option 3 shows the name and library of the exit 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 job is used to locate the program. If no current library entry exists in the library list, QGPL is used.

name Specify the library where the program is located.

Top

Delete option (DLTOPT)

Specifies the action to be taken when:

- A program name is specified on the **Option 3 exit program (EXITPGM)** parameter.
- Option 3 of the programmer menu is selected.
- An object of the name and type to be created already exists in the library specified on the menu.

Regardless of the value specified, the system passes a parameter (from among the parameters passed from the Programmer Menu) to the exit program that specifies whether the object exists.

***DLT** This value must be specified if *NONE is specified on the EXITPGM parameter. If an exit program is specified for the EXITPGM parameter, and the object specified to be created with option 3 exists, and the Enter key is pressed, a message is shown; press the F11 key to proceed. When the F11 key is pressed, the system deletes or replaces the object before calling the program specified by the exit program. This is the normal Programmer Menu function when an exit program is not specified.

NOTES:

1. When the *DLT value is specified, the object is deleted or replaced before the job is submitted or the user exit program is called.
2. If the source type is one of the following, the object is replaced rather than deleted (an exit program must be used to delete instead of replace):

```
BAS       C         CBL
CBL36     CLP       DSPF
DSPF36    FTN       ICFF
MNU36     MSGF36    PAS
PLI       PRTF      RPG
RPG36     RPT36
```

***PROMPT**

The system does not delete or replace the object, but you are prompted for approval to delete the object. If the object exists, and the Enter key is pressed, a message is displayed. Press the F11 key to proceed; the system does not delete the object. The user confirms whether the object is deleted or replaced, yet the deletion is still controlled by the exit program.

***NODLT**

The user exit program is called regardless of the presence of the object.

Table 1. **Figure: Table 1. Actions Taken by the System When an Exit Program Is Called**

DLTOBJ Specified	F11 Required	Object Deleted	Value Passed to Exit Program
If object existed when option 3 was selected			
*DLT	Yes	Yes	1
*PROMPT	Yes	No	0
*NODLT	No	No	0
If object did not exist when option 3 was selected			
*DLT	No	---	2
*PROMPT	No	---	2
*NODLT	No	---	2

Top

Examples

Example 1: Displaying Programmer Menu

```
STRPGMMNU
```

This command displays the Programmer Menu with defaults for all parameters. This has the same result as entering CALL QPGMMENU.

Example 2: Preventing Values from Being Changed

```
STRPGMMNU   SRCFILE(YOURFILE)  SRCLIB(YOURLIB)  OBJLIB(YOURLIB)
            JOB(YOURJOBD)  ALWUSRCHG(*NO)
```

This command prevents the values on the menu from being changed from those specified on the command.

Example 3: Calling an Exit Program

```
STRPGMMNU   EXITPGM(OPT3PGM)  DLTOPT(*PROMPT)
```

This command calls user exit program OPT3PGM instead of submitting a batch job when option 3 is specified. If the object already exists, DLTOPT(*PROMPT) requires the user to press the F11 key; however, the object is not deleted.

Example 4: Receiving Parameters

The following portion of a CL program is an example of how these parameters would be received by a user exit program. If the specified type is one of those listed, the object is not deleted. The create command with REPLACE(*YES) specified is passed to the exit program. The value passed to the exit program is 0.

```
PGM   PARM(&OPTION &PARM &TYPE &PARM2 &SRCFILE  +
           &SRCLIB &OBJLIB &JOBD &RQSLEN &RQSDTA512  +
           &F4 &F11 &EXIST)
/* The following values are passed in exactly as  */
/* they appear on the Programmer Menu.            */
DCL   VAR(&OPTION)   TYPE(*CHAR)  LEN(2)
DCL   VAR(&PARM)     TYPE(*CHAR)  LEN(10)
DCL   VAR(&TYPE)     TYPE(*CHAR)  LEN(10)
DCL   VAR(&PARM2)    TYPE(*CHAR)  LEN(21)
DCL   VAR(&SRCFILE)  TYPE(*CHAR)  LEN(10)
DCL   VAR(&SRCLIB)   TYPE(*CHAR)  LEN(10)
DCL   VAR(&OBJLIB)   TYPE(*CHAR)  LEN(10)
DCL   VAR(&JOBD)     TYPE(*CHAR)  LEN(10)
/* The following values are derived by QPGMMENU     */
/* from the information entered to the above fields */
/* and the F keys.                                  */
/* NUMBER OF BYTES OF REQUEST DATA */
DCL   VAR(&RQSLEN)  TYPE(*DEC)  LEN(3 0)
/* DATA FOR RRQSDTA PARAMETER OF SBMJOB COMMAND. */
DCL   VAR(&RQSDTA512)  TYPE(*CHAR)  LEN(512)
/* F4 WAS PRESSED, '1', OTHERWISE '0'.   */
DCL   VAR(&F4)  TYPE(*CHAR)  LEN(1)
/* F11 WAS PRESSED, '1', OTHERWISE '0'.     */
DCL   VAR(&F11)  TYPE(*CHAR)  LEN(1)
/* OBJECT EXISTS- '0 OBJECT WAS DELETED- '1'
OR OBJECT DID NOT EXIST -'2'*/
DCL   VAR(&EXIST)  TYPE(*CHAR)  LEN(1)
```

Additional information, along with examples of the STRPGMMNU command with the EXITPGM parameter, can be found in the **CL Programming** book in the Information Center.

Top

Error messages

None

Top

Start Program Profiling (STRPGMPRF)

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

Parameters
Examples
Error messages

The Start Program Profiling (STRPGMPRF) command starts collection of profiling information of ILE programs or service programs that have been enabled to collect profiling data by specifying PRFDTA(*COL) on the Change Program (CHGPGM) or Change Service Program (CHGSRVPGM) command, or when the modules were created by a compiler, or by specifying PRFDTA(*COL) on the Change Module (CHGMOD) CL command. All programs compiled or changed with this option that are active will have profiling information updated until an End Program Profiling (ENDPGMPRF) command is issued.

The profiling information is added to existing profiling information. If this is not desired, the profiling data can be cleared by specifying PRFDTA(*CLR) on the CHGPGM or CHGSRVPGM command.

Restrictions:

- This command is shipped with no public (*EXCLUDE) authority, and QPGMR user profile having use (*USE) authority to the command.

There are no parameters for this command.

Top

Parameters

None

Top

Examples

```
STRPGMPRF
```

This command start the collection of program profiling information.

Top

Error messages

***ESCAPE Messages**

CPF5CAA
Unexpected error occurred during program profiling.

Top

Start Prestart Jobs (STRPJ)

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

Parameters
Examples
Error messages

The Start Prestart Jobs (STRPJ) command starts jobs for a prestart job entry in an active subsystem when there are no currently active prestart jobs for the prestart job entry.

This command is valid after an ENDPJ command is complete, or when all prestart jobs have been ended by the system due to an error or were never started during subsystem start up due to STRJOBS (*NO) on the ADDPJE command. The number of jobs started is determined by the INLJOBS value on the prestart job entry.

Restrictions:

1. To use this command, you must have:
   - job control (*JOBCTL) special authority.
   - use (*USE) authority to the subsystem description.
   - use (*USE) authority to the program and execute (*EXECUTE) authority to the library that contains the program.
   - use (*USE) authority to all auxiliary storage pool (ASP) device descriptions in the ASP group if the subsystem description specifies an ASP group.

Top

Parameters

Keyword	Description	Choices	Notes
SBS	Subsystem	*Name*	Required, Positional 1
PGM	Program	*Qualified object name*	Required, Positional 2
	Qualifier 1: Program	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	

Top

Subsystem (SBS)

Specifies the name of the active subsystem that contains the prestart job entry.

This is a required parameter.

name Specify the name of the active subsystem that contains the prestart job entry.

Top

Program (PGM)

Specifies the name of the program that identifies the prestart job entry. This program name is used to match an incoming request.

This is a required parameter.

Qualifier 1: Program

name Specify the name of the program that identifies the prestart job entry.

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 program's library.

Top

Examples

```
STRPJ   SBS(SBS1)  PGM(PJLIB/PJPGM)
```

This command starts prestart jobs for prestart job entry PJPGM in subsystem SBS1. Subsystem SBS1 must be active when this command is issued. The number of jobs started is the number specified in the INLJOBS value of prestart job entry PJPGM. The subsystem starts program PJPGM in library PJLIB.

Top

Error messages

***ESCAPE Messages**

CPF0921
Start Prestart Jobs command not allowed now.

CPF101C
Not authorized to device &1.

CPF1226
Start prestart jobs failed.

CPF1227
No authority has been granted to use command.

CPF1317
No response from subsystem for job &3/&2/&1.

CPF1351
Function check occurred in subsystem for job &3/&2/&1.

CPF1834
Prestart job entry for program &1 in &2 does not exist.

CPF1835
Not authorized to subsystem description.

CPF9810
Library &1 not found.

CPF9811
Program &1 in library &2 not found.

CPF9820
Not authorized to use library &1.

CPF9821
Not authorized to program &1 in library &2.

Top

Start Printer Emulation (STRPRTEML)

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

Parameters
Examples
Error messages

The Start Printer Emulation (STRPRTEML) command starts 3270 printer emulation using a binary synchronous communications (BSC) or Systems Network Architecture (SNA) emulation printer device and a printer device file. The STRPRTEML command is used to print host system (System/370 type) information on an iSeries system. It is used when the user is working on an iSeries system and the information is on a System/370 type system.

More information is in the 3270 Device Emulation Support book, SC41-5408.

Top

Parameters

Keyword	Description	Choices	Notes
EMLCTL	Emulation controller, or	*Name*	Optional, Positional 1
EMLDEV	Emulation device, or	*Name*	Optional, Positional 2
EMLLOC	Emulation location	*Communications name*	Optional, Positional 3
PRTDEV	Print device	*Name*	Optional
JOB	Job name	*Name*, ***EMLDEV**	Optional
ENDBKTEJT	End Bracket eject	***NO**, *YES	Optional
PRTFILE	Printer file	*Qualified object name*	Optional
	Qualifier 1: Printer file	*Name*, **QPEMPRTF**	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
TIMOUTEJT	Timeout wait eject	Single values: ***NONE**, *IMMED Other values: *Element list*	Optional
	Element 1: Minutes	0-99	
	Element 2: Seconds	0-59, **0**	
DFRPRTOUT	Defer printing spool output	***PRTFILE**, *YES, *NO	Optional
SPOOL	Spool output	***PRTFILE**, *YES, *NO	Optional
OPNPRTF	Open printer file	***IMMED**, *RCVDTA	Optional
NUMCOL	Print positions per line	1-378, ***PRTFILE**	Optional
NUMLIN	Lines per page	1-255, ***PRTFILE**	Optional
LPI	Lines per inch	***PRTFILE**, 6, 3, 4, 7.5, 7,5, 8, 9	Optional
MSGQ	Message queue	Single values: ***DSPDEV**, *NONE Other values: *Qualified object name*	Optional
	Qualifier 1: Message queue	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	

Keyword	Description	Choices	Notes
CHRSET	Language character set	***SYSVAL**, AGB, AGE, AGI, AGM, ALI, ALM, BGB, BGE, BLI, BLM, BRB, BRE, CAB, CAE, CAI, CAM, CLB, CLE, CYB, CSB, CSE, DMB, DMI, DME, DMM, ESB, FAB, FAI, FAE, FAM, FNB, FNI, FQB, FQI, FNE, FNM, ICE, GKB, GNB, GNE, HIB, HNB, HNE, ICB, ICI, ICM, INB, INI, IRB, ITB, ITE, ITI, ITM, JEB, JEI, JKB, JPB, JPE, JUB, KAB, KOB, LTB, LAE, LVB, MKB, MKE, NCB, NCE, NEB, NEI, NEE, NEM, NWB, NWE, NWI, NWM, PKE, PLB, PLE, PRB, PRI, PRE, PRM, RCB, RMB, RME, ROB, ROE, RUB, RUE, SFI, SFM, SGM, SGI, SKB, SKE, SPB, SPE, SPI, SPM, SQB, SQE, SSB, SSI, SWB, SWI, SSE, SSM, SWE, SWM, TAB, THB, THE, TKB, TKE, TRB, TRE, UAE, UKB, UKI, USB, USI, UKE, UKM, USE, USM, VNE, YGI, YGM, *TRNTBL	Optional
SBMJOB	Submit job	***YES**, *NO	Optional
JOBD	Job description	*Qualified object name*	Optional
	Qualifier 1: Job description	*Name*, **QBATCH**	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
TRNTBLOUT	Outgoing translation table	Single values: ***CHRSET** Other values: *Qualified object name*	Optional
	Qualifier 1: Outgoing translation table	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
JOBQ	Job queue	Single values: ***JOBD** Other values: *Qualified object name*	Optional
	Qualifier 1: Job queue	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
ENDCOND	End emulation conditions	Single values: ***NONE** Other values (up to 3 repetitions): *DACTLU, *ENDBKT, *UNBIND	Optional
FORMFEED	Acknowledge form feed	***YES**, *NO	Optional
EMLCFGE	Configuration entry	*Name*, QEMDFTCFGE, ***NONE**	Optional

Top

Emulation controller (EMLCTL)

Specifies the name of a BSC controller description or SNA controller description that has attached 3270 printer emulation device descriptions. When this parameter is specified, the printer emulation job uses a 3270 printer emulation device attached to this controller description. The requester must be authorized to the controller and at least one device, and the device must be available.

Either this parameter, the **Emulation device (EMLDEV)** parameter, or the **Emulation location (EMLLOC)** parameter is required.

Top

Emulation device (EMLDEV)

Specifies the name of a BSC or an SNA printer emulation device (EMLDEV(3284, 3286, 3287, 3288, or 3289)) that is used by the printer emulation job to do a type 3270 printer emulation. The user must be authorized to the device, and the device must be available.

Top

Emulation location (EMLLOC)

Specifies the emulation remote location name that describes the location of the 3270 printer emulation devices. This name is defined during device description configuration and refers to the remote location where communication takes place. When this parameter is specified, an available printer emulation device is selected from those referred to by the location. At least one printer emulation device referred to by the location must be available, and the job running emulation must be authorized to use the device.

Either this parameter, the **Emulation controller (EMLCTL)** parameter, or the **Emulation device (EMLDEV)** parameter is required.

Top

Print device (PRTDEV)

Specifies the name of the printer used with this printer device file to print the output. This parameter is ignored if the printer data is spooled. If the output is not spooled and the printer device is being used at the time the job is initiated, the emulation session ends.

Note: If a printer name is not specified, the output is sent to the printer device specified on the **Device (DEV)** parameter of the Create Printer File (CRTPRTF) command. This printer device can be displayed by using the Display File Description (DSPFD) command.

Top

Job name (JOB)

Specifies a job name for the printer emulation job. If the EMLDEV parameter is not specified, and a batch job is to be submitted (*YES is specified for the **Submit job (SBMJOB)** parameter), a job name must be specified. This parameter is ignored when *NO is specified for the SBMJOB parameter.

***EMLDEV**
The job name is the same as the printer emulation device name.

job-name
Specify a name for this printer emulation job.

Top

End Bracket eject (ENDBKTEJT)

Specifies whether SNA printer emulation should force out the emulation output when an SNA End Bracket (EB) is received from the host system. Emulation output is forced out by closing and then reopening the emulation printer file specified in the PRTFILE parameter. When the emulation printer output is ejected, a page eject is performed. The parameter uses the default value of *NO for BSC printer emulation.

***NO** The emulation output is not forced out when SNA printer emulation receives an End Bracket.

***YES** The emulation output is forced out when SNA printer emulation receives an End Bracket. This is done only if the open printer file contains host system data.

Top

Printer file (PRTFILE)

Specifies the printer device file that prints data received from the host system. The printer device file can be spooled or not spooled.

QPEMPRTF
The standard printer file (which specifies SPOOL(*YES)) shipped with the emulation program is used as the printer device file.

printer-device-file-name
Specify the name and library of a user-defined printer device 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 printer device file. If no library is specified as the current library for the job, QGPL is used.

library-name
Specify the library where the printer device file is located.

Top

Timeout wait eject (TIMOUTEJT)

Specifies whether printer emulation forces out the emulation output when a time-out has occurred while waiting for host system data. The emulation output is forced out only if the open printer file contains host system data. The wait interval can be specified in number of minutes or seconds. If a value is specified for both minutes and seconds, then these values will be added together.

Note: When the emulation printer output is ejected, a page eject is also performed.

***NONE**
The emulation output is not forced out based on a specified timeout period.

***IMMED**
The emulation output is forced out immediately.

minutes-seconds
Specify a time-out wait interval in minutes or seconds or both.

Minutes
Valid values range from 0 through 99.

Seconds
Valid values range from 0 through 59.

Top

Defer printing spool output (DFRPRTOUT)

Specifies whether spooled output is printed immediately or is delayed. This parameter is ignored when *NO is specified for the Spool output (SPOOL) parameter.

***PRTFILE**
The SCHEDULE value for the printer file controls how the spooled output is printed.

***YES** Spooled output is printed when the spooled file is closed.

***NO** Spooled output can be printed before the spooled file is closed. The printed output does not contain all the data sent by the host system until the spooled file is closed. If the printer is not using spooling, this parameter is ignored.

Note: Once the printer starts printing output from 3270 device emulation, spooled output from other jobs sharing the printer does not print until the spooled file that is currently printing is complete.

Top

Spool output (SPOOL)

Specifies whether the output data for the printer device file is spooled. If *NO is specified on this parameter, output is sent to the printer specified on the **Print device (PRTDEV)** parameter. If the output is spooled, it is sent to the output queue specified on the **Spooled output queue (OUTQ)** parameter of the Create Printer File (CRTPRTF) command for the printer file specified on the PRTFILE parameter. The Output Queue value can be displayed by using the Display File Description (DSPFD) command and specifying the printer file name.

Note: If *JOB is specified on the OUTQ parameter, output is sent to the output queue specified on the OUTQ parameter of the job description specified on the JOBD parameter. This value can be displayed by using the Display Job Description (DSPJOBD) command and specifying the job description name.

***PRTFILE**
The value specified on the **Spool output (SPOOL)** parameter of the Create Printer File (CRTPRTF) command determines whether spooling is performed.

***YES** The data is spooled.

***NO** The data is not spooled; it is sent directly to the device and is printed as the output becomes available.

Top

Open printer file (OPNPRTF)

Specifies when the printer file is opened during the SNA 3270 printer emulation session. If the printer data is not spooled, then the printer will be allocated to your job when the printer file is opened. If the printer data is spooled, then the spool writer is allocated to your job after the printer file is opened depending on the value of the DFRPRTOUT parameter.

This parameter is not allowed if specified for BSC 3270 printer emulation.

***IMMED**
The printer file is opened immediately after starting the 3270 printer emulation session.

***RCVDTA**
The printer file is opened after first receiving print data from the host system.

Top

Print positions per line (NUMCOL)

Specifies the number of columns in a line when creating the printed output.

***PRTFILE**
The printer file PAGESIZE(width) value contains the number of columns per line. This value is used if *IMMED is specified on the open printer file (OPNPRTF) parameter, or *RCVDTA is specified on the OPNPRTF parameter and the maximum print positions (MPP) value is not sent from the host system. Otherwise, the MPP value sent from the host system is used.

number-of-columns
Specify the number of columns per line in the printed output. Valid values range from 1 through 378.

Top

Lines per page (NUMLIN)

Specifies the number of lines per page when creating the printed output.

***PRTFILE**
The printer file PAGESIZE(length) value contains the number of lines per page. This value is used if *IMMED is specified on the open printer file (OPNPRTF) parameter, or *RCVDTA is specified on the OPNPRTF parameter and the maximum page length (MPL) value is not sent from the host system. Otherwise, the MPL value sent from the host system is used.

number-of-lines
Specify the number of lines per page in printed output. Valid values range from 1 through 255.

Top

Lines per inch (LPI)

Specifies the number of lines per inch when creating the printed output.

***PRTFILE**
The printer file LPI value contains the number of lines per inch. This value is used if *IMMED is specified on the open printer file (OPNPRTF) parameter, or *RCVDTA is specified on the OPNPRTF parameter and the set line density (SDL) value is not sent from the host system. Otherwise, the SDL value sent from the host system is used.

lines-per-inch
Specify the number of lines per inch in the printed output. Valid values are 3, 4, 6, 7.5, 7,5, 8 and 9. Values 3, 7.5 and 7,5 are valid only for double-byte character set (DBCS) printer devices.

Top

Message queue (MSGQ)

Specifies the message queue to which operational messages for this device are sent.

***DSPDEV**
The current display station message queue is used.

***NONE**
No messages are sent to message queues other than the job log for the printer emulation job.

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 message queue. If no library is specified as the current library for the job, QGPL is used.

library-name
Specify the library where the message queue is located.

message-queue-name
Specify the name and library of the message queue where messages are sent when the printer emulation job is running.

Top

Language character set (CHRSET)

Specifies the 3-character country or region keyboard language identifier which represents a specific full character identifier (CHRID - comprised of a character set and code page) that is used by the printer being emulated.

***SYSVAL**
The current QKBDTYPE system value is used.

country-keyboard-identifier
Specify the country or region keyboard language identifier to be used.

***TRNTBL**
Allows a user-defined translation table to be used. The character translation is defined in the translation table specified by the **Outgoing translation table (TRNTBLOUT)** parameter. The CHRID associated with the QKBDTYPE system value will be used by the printer being emulated.

Top

Submit job (SBMJOB)

Specifies whether the printer emulation should be done as a separate job or as part of this job.

***YES** A specific job is submitted to do the printer emulation. The job attributes are determined from the job description specified by the **Job description (JOBD)** parameter. The job uses your user profile.

***NO** Printer emulation is done in the current job.

Top

Job description (JOBD)

Specifies the job description for the job that is being submitted for 3270 printer emulation. This parameter is ignored when *NO is specified for the **Submit job (SBMJOB)** parameter.

QBATCH
The job description QBATCH is used for the job.

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 job description name. If no library is specified as the current library for the job, QGPL is used.

library-name
Specify the library where the job description name is located.

job-description-name
Specify the name and library of the job description associated with the job.

Top

Outgoing translation table (TRNTBLOUT)

Specifies the outgoing translation table used to translate characters sent from the host system to 3270 Emulation.

***CHRSET**
Specify that translation is done when data is sent from the host system using the character set specified on the **Language character set (CHRSET)** 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 translation table. If no current library entry exists in the library list, QGPL is used.

library-name
Specify the library where the translation table is located.

table-name
Specifies the name and library of the table which is used for outgoing translation.

Top

Job queue (JOBQ)

Specifies the name of the job queue in which this 3270 printer emulation job is placed. This parameter is ignored when *NO is specified for the **Submit job (SBMJOB)** parameter.

***JOBD**
The submitted job is placed in the job queue associated with the job description specified in the (JOBD) parameter.

job-queue-name
Specify the name and library of the job queue to contain the submitted job.

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

library-name
Specify the library where the job queue is located.

Top

End emulation conditions (ENDCOND)

Specifies additional ways in which the SNA 3270 printer emulation session can end.

This parameter is not allowed if specified for BSC 3270 printer emulation.

***NONE**
No additional ways to end 3270 printer emulation are requested.

***DACTLU**
The 3270 printer emulation session ends if it receives an SNA DACTLU from the host system.

***ENDBKT**
The 3270 printer emulation session ends if it receives an SNA end bracket from the host system. Please consider the following before selecting this end condition:

- This end condition should be used only when you need to print one host system file for the duration of the session. An end bracket may occur after printing the first file, and the 3270 session ends before a second file can print.

***UNBIND**
The 3270 printer emulation session will end if it receives an SNA UNBIND from the host system. Please consider the following before selecting this end condition:

- This end condition should be used only when you need to print one host system file for the duration of the session. An UNBIND may occur after printing the first file, and the 3270 session will end before a second file can print.

Top

Acknowledge form feed (FORMFEED)

Specifies whether to acknowledge a form-feed instruction located in the first character position of the first print line for a 3270 Information Display System data-stream compatibility (DSC) LU3 printer.

This parameter is ignored for an SNA character string (SCS) LU1 printer.

***YES** The form-feed instruction is acknowledged. The print position advances to a new page.

***NO** The form-feed instruction is ignored. The print position does not advance to a new page.

Top

Configuration entry (EMLCFGE)

Specifies whether a configuration entry is used for this session. Configuration entries indicate 3270 emulation configuration options. Configuration entries are created with the Add Emulation Configuration Entry (ADDEMLCFGE) command.

***NONE**
No configuration entry is named and the configuration entry defaults are used.

QEMDFTCFGE
The default configuration entry QEMDFTFGE is used. This entry is shipped with configuration entry defaults, and can be updated with the Change Emulation Configuration Entry (CHGEMLCFGE) command.

configuration-entry-name
Specify the name of the configuration entry to be used. If the configuration entry named does not exist in the configuration file, the configuration entry defaults are used.

Top

Examples

Example 1: Printing Data to Standard Emulation Printer File

```
STRPRTEML   EMLDEV(HOSTPRT4)
```

This command starts a batch job by accepting data from the HOSTPRT4 device and prints the data to the standard emulation printer file (QPEMPRTF). The job is named HOSTPRT4 and runs until the job is canceled. Messages are sent to the current work station message queue.

Example 2: Emulating a Printer in the Current Job

```
STRPRTEML   EMLDEV(HOSTPRT5)  SBMJOB(*NO)
```

This command does printer emulation in the current job by accepting data from the HOSTPRT5 device, and writing the data to the standard emulation printer device file (QPEMPRTF). The request is active until it ends through the End Printer Emulation (ENDPRTEML) command, or until the job is canceled.

Example 3: Printing Output Immediately

```
STRPRTEML   EMLCTL(EMLCTL1)  TIMOUTEJT(10)
            DFRPRTOUT(*NO)  NUMLIN(96)
```

This command starts a batch job by accepting data from the device and printing the data in printer file QPEMPRTF. If a timeout of 10 minutes occurs, printer emulation forces out the emulation output. The output prints immediately; the maximum number of lines per page is 96.

Top

Error messages

***ESCAPE Messages**

CPF85EB
3270 device emulation session ended.

CPF85ED
Values other than ENDCOND(*NONE) are not supported.

CPF85EE
*RCVDTA on the OPNPRTF parameter is not supported.

CPF8510
Internal error occurred on device &1.

CPF8511
Emulation ended by errors on device &2.

CPF8512
Emulation ended because device &2 was held.

CPF8513
Emulation ended by errors on device &2.

CPF8514
Error recovery stopped on device &1.

CPF8515
3270 emulation session ended by host.

CPF8516
No match between host and device &2.

CPF8517
Received more than maximum number of fields allowed.

CPF8518
Emulation ended because of internal failure in system.

CPF8519
Function check in 3270 emulation.

CPF8561
Printer emulation not started.

CPF8564
Printer emulation job &3/&2/&1 ended.

CPF8570
Translate of 3270 printer data stream failed.

CPF8579
Cannot open printer file &1 in library &2.

CPF8580
File &1 is not a printer file or has been overridden.

CPF8582
Cannot open printer file &1 again in library &2.

CPF8583
Printer emulation cannot open required file.

Top

Start Printer Writer (STRPRTWTR)

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

Parameters
Examples
Error messages

The Start Printer Writer (STRPRTWTR) command starts a spooling writer to the specified printer. The writer, which is a system job, takes spooled files from an output queue and produces (writes) the output on the printer device. This command specifies the name of the printer, the names of the output and message queues used, and the name of the writer.

More than one writer can be active at the same time (as determined by the spooling subsystem description), and up to 10 writers can be active to the same output queue. Each writer must have a unique writer name, its own device, and only one type of writer (print, remote, or diskette) can be active to a single output queue. A writer that has been started can be actively writing output or waiting for a file entry to be put on the output queue. The writer can be changed, held, or canceled if the Change Writer (CHGWTR), Hold Writer (HLDWTR), or End Writer (ENDWTR) command is used. Because each writer runs independently of the job that started it, you can continue doing other work on the system after you start a writer.

Top

Parameters

Keyword	Description	Choices	Notes
DEV	Printer	*Name*, *ALL, *SYSVAL	Required, Positional 1
OUTQ	Output queue	Single values: ***DEV** Other values: *Qualified object name*	Optional, Positional 2
	Qualifier 1: Output queue	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
MSGQ	Queue for writer messages	Single values: ***DEVD**, *REQUESTER Other values: *Qualified object name*	Optional, Positional 4
	Qualifier 1: Queue for writer messages	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
FORMTYPE	Form type options	*Element list*	Optional
	Element 1: Form type	*Character value*, ***ALL**, *STD, *FORMS	
	Element 2: Message option	***INQMSG**, *MSG, *NOMSG, *INFOMSG	
FILESEP	File separators	0-9, ***FILE**	Optional
SEPDRAWER	Drawer for separators	1-255, ***DEVD**, *FILE	Optional
WTR	Writer	*Name*, ***DEV**	Optional, Positional 3
AUTOEND	Auto-end options	*Element list*	Optional, Positional 5
	Element 1: Automatically end writer	***NO**, *YES	
	Element 2: If yes, when to end	***NORDYF**, *FILEEND	
ALWDRTPRT	Allow direct print	***NO**, *YES	Optional
ALIGN	Align page	*FILE, ***WTR**, *FIRST	Optional

Keyword	Description	Choices	Notes
INIT	Initialize printer	***WTR**, *FIRST, *ALL	Optional
FILE	Spooled file	*Name*, ***NONE**, *LAST	Optional
JOB	Job name	Single values: ***** Other values: *Qualified job name*	Optional
	Qualifier 1: Job name	*Name*	
	Qualifier 2: User	*Name*	
	Qualifier 3: Number	000000-999999	
SPLNBR	Spooled file number	1-999999, ***ONLY**, *LAST, *ANY	Optional
JOBSYSNAME	Job system name	*Name*, ***ONLY**, *CURRENT, *ANY	Optional
CRTDATE	Spooled file created	Single values: ***ONLY**, *LAST Other values: *Element list*	Optional
	Element 1: Creation date	*Date*	
	Element 2: Creation time	*Time*, ***ONLY**, *LAST	
PAGE	Starting page	*Integer*, ***BEGIN**	Optional

Top

Printer (DEV)

Specifies the printer device used to print the spooled file. The device must be available before the writer can be started.

This is a required parameter.

***ALL** Start a printer writer for every printer configured on the system.

***SYSVAL**
Start a printer writer for the system default printer.

name Specify the name that the printer device being started is identified by.

Top

Output queue (OUTQ)

Specifies the output queue from which the writer processes spooled files.

Single values

***DEV** Use the default output queue associated with the printer specified on the **Printer (DEV)** parameter.

Qualifier 1: Output queue

name Specify the name of the output queue that the writer processes spooled files from.

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 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 located.

Top

Queue for writer messages (MSGQ)

Specifies the message queue that is used by this user.

Single values

***DEVD**
The messages are sent to the message queue specified in the printer's device description.

***REQUESTER**
The messages are sent to the workstation message queue of the workstation of the user who started the process. If this value is specified for a batch job, *DEVD is used.

Qualifier 1: Queue for writer messages

name Specify the name of the message queue where writer messages should be 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 job is used to locate the message queue. If no current library entry exists in the library list, QGPL is used.

name Specify the name of the library where the message queue is located.

Top

Form type options (FORMTYPE)

Specifies which form type should be selected to print. This parameter specifies that only the files with this form type are processed now. All other files are left on the output queue as available. If you wish to change the type of form to be printed after the writer is started, you should use the Change Writer (CHGWTR) command.

Note: The form load message is issued when the spooled file to be printed has a form type different from the form type of the last spooled file that was printed on the device. The last form type printed is kept from the last STRPRTWTR, CHGWTR, or VRYCFG command issued.

Consider the following example:

1. The last spooled file printed on printer PRT01 had the form type *STD.
2. The user changes the form type on PRT01 to XYZ using the following command:
   ```
   CHGWTR  PRT01  FORMTYPE(XYZ)
   ```
3. No spooled file with the form type XYZ is printed on PRT01.
4. The user then sends a spooled file with the form type *STD to PRT01. The form load message is not issued, despite the intervening CHGWTR command, because the last spooled file printed on PRT01 had the same form type as the spooled file being printed.

   The form load message would be issued if a spooled file with the form type XYZ were actually printed on PRT01.

Element 1: Form type

***ALL** All available files on the output queue will be processed regardless of their form type.

***FORMS**
All available files on the output queue with the same form type are processed as a group before the writer moves on to the next form type. The writer first chooses the first available file on the queue. After the first file is complete, all files with the same form type will be processed. The writer again chooses the first available file on the queue and repeats the process for that form type.

***STD** The writer processes spooled files with a form type of *STD.

form-type
Specify the type of form for which you want spooled files processed.

Element 2: Message option

***INQMSG**
An inquiry message is sent to the message queue when a spooled file has a form type that is different than the form type in the printer.

***INFOMSG**
An informational message is sent to the message queue when no spooled files requiring this form type remain in the output queue.

***MSG** An inquiry message is sent to the message queue when a spooled file has a form type that is different than the form type in the printer and an informational message is sent when no spooled files requiring this form type remain in the output queue.

***NOMSG**
Neither an inquiry message nor an informational message is sent to the message queue.

Top

File separators (FILESEP)

Specifies how to control the number of file separator pages that are printed before each file.

***FILE** The number of separators that was specified for each individual file is used.

0-9 Specify the number of separator pages to print. Whenever you respond to the change form type message indicating that a new form type has been put on the printer, the writer issues a message inquiring how many file separator pages should be printed with the new form type.

Top

Drawer for separators (SEPDRAWER)

Specifies which paper drawer is selected for printing job and file separators.

***DEVD**
The value stored in the device description for the printer is used.

***FILE** The separator pages are printed from the same drawer as the spooled file.

1 The separator pages are printed from drawer 1.

2 The separator pages are printed from drawer 2.

3 The separator pages are printed from drawer 3.

1-255 Specify a number to indicate the drawer from which the separator pages are printed.

Note: For some printers, SEPDRAWER(3) implies an envelope drawer.

Top

Writer (WTR)

Specifies the name of the spooling writer being started. Each writer name must be unique.

***DEV** The name of the writer is the same as the name of the printer device specified on the **Printer (DEV)** parameter.

name Specify the name by which the writer being started is identified.

Top

Auto-end options (AUTOEND)

Specifies whether the writer ends automatically.

Element 1: Automatically end writer

***NO** The writer does not end when the last available file has been removed from the output queue; it waits for another spooled file entry to be put on the queue.

***YES** The writer automatically ends after it has reached the state specified by the second part of this parameter.

Element 2: If yes, when to end

***NORDYF**
The writer automatically ends when there are no ready files (all the available files have been removed from the output queue).

***FILEEND**
The writer stops after it has finished processing one spooled file.

Top

Allow direct print (ALWDRTPRT)

Specifies whether the printer writer allows files to be printed directly to the printer. A file printed directly to the printer is created by specifying SPOOL(*NO) for a printer file. When direct printing is allowed, the non-spooled printer file is printed immediately if the printer is available or, if the printer is busy, the non-spooled printer file waits until the printer is available. The maximum wait is the length of time specified on the WAITFILE parameter on the printer file, after which the job is automatically canceled. The user can cancel a non-spooled printer file only with an End Job (ENDJOB) command.

***NO** The printer does not allow non-spooled printer files to be printed to the device.

***YES** The printer can be used to print spooled and non-spooled output. See the Create, Change, or Override Printer File (CRTPRTF, CHGPRTF, or OVRPRTF) command to set the value of the WAITFILE parameter.

Note: Nonspooled files wait up to 30 seconds regardless of whether the value specified on the WAITFILE parameter is less than 30 seconds.

Top

Align page (ALIGN)

Specifies how to control the forms alignment.

***WTR** The writer keeps track of the output that is printed and issues a forms alignment message whenever it determines that forms may need to be aligned.

***FILE** The forms alignment message is issued for every file that has *YES specified for the **Align page** prompt (ALIGN parameter). This option should be taken whenever the automatic forms alignment control provided by the writer does not provide the desired results.

***FIRST**
The forms alignment message is issued only for the first file printed. No alignment messages are issued when subsequent errors occur on the printer.

Top

Initialize printer (INIT)

Specifies how often to initialize the printer device.

Note: This parameter is ignored if TRANSFORM(*YES) or a user data transform program was specified on the printer device description.

***WTR** The writer initializes the printer device when necessary.

***FIRST**
The writer initializes the printer device only before the first file is printed, or after a device error occurs.

***ALL** The writer initializes the printer device before each file and each copy of the file is printed.

Top

Spooled file (FILE)

Specifies the first (or only) spooled file to print on the printer. If several files are available on the output queue, the next file produced is the first one available with the highest priority.

***NONE**
No spooled file name is specified; the first spooled file that becomes available on the output queue is processed first.

***LAST**
The spooled file which was being printed when the writer ended will start to print when the writer is restarted.

name Specify the name of the spooled file that is the first (or only) file to be written to the printer.

Top

Job name (JOB)

Specifies the name of the job that created the spooled file. This parameter is valid only if a spooled file name is specified on the **Spooled file (FILE)** parameter.

Single values

*****
The job from which this Start Printer Writer (STRPRTWTR) command was issued is the job that created the spooled file.

Qualifier 1: Job name

name Specify the name of the job that created the spooled file. If no other job qualifiers are given, all of the jobs currently in the system are searched for the simple name of the job.

Qualifier 2: User

name Specify the user name that identifies the user profile under which the job is run.

Qualifier 3: Number

000000-999999
Specify the system-assigned job number.

Top

Spooled file number (SPLNBR)

Specifies the number of the spooled file that is processed first. This parameter is valid only if a spooled file name is specified on the **Spooled file** prompt (FILE parameter).

***ONLY**
Only one spooled file in the job has the specified file name; therefore, the number of the spooled file is not necessary.

***LAST**
The spooled file with the highest number and the specified file name is used.

***ANY** The spooled file number is not used to determine which spooled file is used. Use this value when the job system name parameter or the spooled file create date and time parameter is to take precedence over the spooled file number when selecting a spooled file.

1-999999
Specify the number of the specified file from the job on the specified output queue that is to be processed first.

Top

Job system name (JOBSYSNAME)

Specifies the name of the system where the job that created the spooled file (JOB parameter) ran. This parameter is considered after the job name, user name, job number, spooled file name, and spooled file number parameter requirements have been met.

***ONLY**
There is one spooled file with the specified job name, user name, job number, spooled file name, spooled file number, and spooled file create date and time.

***CURRENT**
The spooled file created on the current system with the specified job name, user name, job number, spooled file name, spooled file number, and create date and time is used.

***ANY** The job system name is not used to determine which spooled file is used. Use this value when the spooled file create date and time parameter is to take precedence over the job system name when selecting a spooled file.

name Specify the name of the system where the job that created the spooled file ran.

Top

Spooled file created (CRTDATE)

Specifies the date and time the spooled file was created. This parameter is considered after the job name, user name, job number, spooled file name, spooled file number, and job system name parameter requirements have been met.

Single values

***ONLY**
There is one spooled file with the specified job name, user name, job number, spooled file name, spooled file number, and job system name.

***LAST**
The spooled file with the latest create date and time of the specified job name, user name, job number, spooled file name, spooled file number, and job system name is used.

Element 1: Creation date

date Specify the date the spooled file was created.

Element 2: Creation time

***ONLY**
There is one spooled file with the specified job name, user name, job number, spooled file name, spooled file number, job system name, and spooled file create date.

***LAST**
The spooled file with the latest create time of the specified job name, user name, job number, spooled file name, spooled file number, job system name, and spooled file create date is used.

time Specify the time the spooled file was created.

Top

Starting page (PAGE)

Specifies the page number of the first page to print from the first file. This parameter is valid only if a spooled file name is specified on the **Spooled file** prompt (FILE parameter).

***BEGIN**
The restart page of the spooled file is the first page to print.

Note: If this value has not been changed using the Change Spooled File Attribute (CHGSPLFA) command, the file will print in the same manner as it would if the value were *BEGIN.

integer
Specify the number of the first page to print. This number must be within the page range of the file to be valid.

Top

Examples

```
STRPRTWTR   DEV(QSYSPRT)  OUTQ(QPRINTS)  WTR(TOM)
```

This command starts a spooling writer named TOM. This writer takes the output from the output queue named QPRINTS and prints the output on the printer named QSYSPRT. Writer messages are sent to the system operator's message queue, and the writer waits for more output when the queue is emptied.

Top

Error messages

***ESCAPE Messages**

CPF0906
A duplicate job named &3/&2/&1 was found.

CPF1338
Errors occurred on SBMJOB command.

CPF1764
Writer already started for device &1.

CPF1842
Cannot access system value &1.

CPF2115
Object &1 in &2 type *&3 damaged.

CPF2207
Not authorized to use object &1 in library &3 type *&2.

CPF3303
File &1 not found in job &5/&4/&3.

CPF3305
Output queue &1 in library &2 assigned to another writer.

CPF3309
No files named &1 are active.

CPF3310
Writer &1 already started.

CPF3330
Necessary resource not available.

CPF3340
More than one file with specified name found in job &5/&4/&3.

CPF3342
Job &5/&4/&3 not found.

CPF3343
Duplicate job names found.

CPF3347
Device &1 not found.

CPF3357
Output queue &1 in library &2 not found.

CPF3362
Objects in QTEMP not valid for parameter values.

CPF3363
Message queue &1 in library &2 not found.

CPF3369
Device &1 not printer device.

CPF336B
Not found or not authorized to driver exit program &1 in library &2.

CPF3418
Duplicate file &1 number &2 found in job.

CPF346A
Transform exit program &1 in library &2 not found or user is not authorized.

CPF346B
Driver exit program &1 in library &2 not found or user is not authorized.

CPF3463
Output queue for device &1 not found.

CPF3464
Not authorized to output queue &1 in library &2.

CPF3478
File &1 not found in job &5/&4/&3 on output queue &6 in library &7.

Top

Start Query Management Proc (STRQMPRC)

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

Parameters
Examples
Error messages

The Start Query Management Procedure (STRQMPRC) command allows you to run a query management procedure that was saved as a member in a source file.

Top

Parameters

Keyword	Description	Choices	Notes
SRCMBR	Source member	*Name*	Required, Positional 1
SRCFILE	Source file	*Qualified object name*	Optional, Positional 2
	Qualifier 1: Source file	*Name*, **QQMPRCSRC**	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
RDB	Relational database	*Simple name*, ***NONE**, *CURRENT	Optional
RDBCNNMTH	Connection Method	***DUW**, *RUW	Optional
USER	User	*Name*, ***CURRENT**	Optional
PASSWORD	Password	*Character value*, *NONE	Optional
NAMING	Naming convention	***SYS**, *SQL, *SAA	Optional
ALWQRYDFN	Allow information from QRYDFN	***NO**, *YES, *ONLY	Optional
CMDSRCFILE	Command source file	Single values: ***NONE** Other values: *Qualified object name*	Optional
	Qualifier 1: Command source file	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
CMDSRCMBR	Source member	*Name*, ***FIRST**	Optional
ALWDSPLAY	Display screens	***YES**, *NO	Optional

Top

Source member (SRCMBR)

Specifies the source file member that contains the query management procedure to be run.

This is a required parameter.

name Specify the name of the member.

Top

Source file (SRCFILE)

Specifies the source file that contains the query management procedure to be run.

This is a required parameter.

Qualifier 1: Source file

QQMPRCSRC
IBM-supplied source file QQMPRCSRC is used.

name Specify the name of the source 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.

Top

Relational database (RDB)

Specifies the relation database that is accessed during the processing of this command.

***NONE**
The local database is accessed. If the user is connected to a remote database, the connection is reset to local and remains local until completion of this command.

***CURRENT**
The relational database to which the user is currently connected is accessed.

name Specify the name of the relational database that is accessed. The database must have an entry in the relation database directory.

Top

Connection Method (RDBCNNMTH)

Specifies the connection method to use.

***DUW**
Connections to several relational databases are allowed. Consecutive CONNECT statements to additional relational databases do not result in disconnection of previous connections.

***RUW** Only one connection to a relational database is allowed. Consecutive CONNECT statements result in the previous connections being disconnected before a new connection is established.

Top

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.

Top

Password (PASSWORD)

Specifies the password to be used on the remote system.

***NONE**
No password is sent. The user name specified on the USER parameter is not valid if this value is specified.

character-value
Specify the password of the user name specified on the USER parameter.

Top

Naming convention (NAMING)

Specifies the naming convention used for naming objects.

***SYS** The system naming convention is used (name/object-name).

***SQL** The SQL naming convention is used (database-name.object-name). If NAMING(*SQL) is specified, CMDSRCFILE(*LIBL) cannot be specified or allowed as a default value for locating any of the objects specified on other parameters on this command.

***SAA** The SQL naming convention is used (database-name.object-name). If NAMING(*SAA) is specified, CMDSRCFILE(*LIBL) cannot be specified or allowed as a default value for locating any of the objects specified on other parameters on this command.

Top

Allow information from QRYDFN (ALWQRYDFN)

Specifies whether query or form information is taken from a query definition (QRYDFN) object when no query management query (QMQRY) or query management form (QMFORM) object can be found using the specified object name. Any information that has to be derived in this way is discarded when the common programming interface (CPI) command in the procedure is completed. No query management objects are created.

***NO** The information is not taken from a QRYDFN object.

***YES** The information is taken from a QRYDFN object when the specified QMQRY or QMFORM object is not found.

***ONLY**
Information can be derived only from a QRYDFN object. Query management objects are ignored.

Top

Command source file (CMDSRCFILE)

Specifies the command source file that query management uses to run a command procedure. A command procedure can only contain query management set commands which can set application variables as well as query management variables that start with the ’DSQ’ value.

The supported DSQ variables are:

- DSQCMTLV
- DSQCONFIRM
- DSQOAUTH
- DSQSDBNM
- DSQSRUN

Single values

***NONE**
A command source file is not used. The CMDSRCMBR parameter is ignored.

Qualifier 1: Command source file

name Specify the name of the command source 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.

Top

Source member (CMDSRCMBR)

Specifies the command source member that query management uses to run a command procedure. A command procedure can only contain query management set commands which set variables that start with the ’DSQ’ value.

***FIRST**
The first member is used.

name Specify the name of the command source member.

Top

Display screens (ALWDSPLAY)

Specifies the display mode used. The query management session is set to interactive mode if ALWDSPLAY(*YES) is specified. If ALWDSPLAY(*NO) is specified, then the session is set to batch mode. The mode is automatically set to batch when running this command in a batch environment.

***YES** Displays are shown when used in an interactive session. This mode allows you to interact with the query management commands in the procedure.

***NO** No displays are shown.

Top

Examples

Example 1: Running a Query Management Procedure

```
STRQMPRC   SRCMBR(MYPROC)  SRCFILE(RPTLIB/PROCFILE)
```

This command starts the query management procedure stored as the member named MYPROC in the source file named PROCFILE in the RPTLIB library.

Example 2: Taking Information From QRYDFN Objects

```
STRQMPRC   SRCMBR(MYPROC)  SRCFILE(PROCFILE)
           ALWQRYDFN(*YES)  ALWDSPLAY(*NO)
```

This command starts the query management procedure stored as the member named MYPROC in the first file named PROCFILE in the library list for the job. Query and form information is allowed to be taken, as needed, from QRYDFN objects when the procedure statements are processed. No reports are shown but they can be printed if the user specifies a print request. Objects are replaced without confirmation if confirmation is not requested by the user. The procedure ends with some errors if processing locates a global variable that is not set or if confirmation was requested before replacing objects that already exist.

Top

Error messages

***ESCAPE Messages**

QWM2701
&1 command failed.

QWM2703
&1 command ended.

QWM2707
*LIBL not allowed when SQL naming applied.

QWM2709
User or password not valid with relational database value.

QWM2710
Password value *NONE only valid with user value *CURRENT.

QWM2712
Character in user name not valid.

Top

Start Query Management Query (STRQMQRY)

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

Parameters
Examples
Error messages

The Start Query Management Query (STRQMQRY) command is used to run a query.

To use this command, you must first identify the query that is to be processed. The query is any single Structured Query Language (SQL) statement in a QMQRY object. The SQL statement can also be taken from a query definition (QRYDFN) object when a QMQRY object does not exist.

You can show the output on the display, print it, or store it in a database file.

If the SQL statement inside the query does not create an answer-set, then no report or output file is created. This happens if the SQL statement inside a query is not valid or the SQL statement is not a SELECT clause.

If the query contains substitution variables, the SETVAR parameter can be used to set the variables for the query. If prompting is enabled, query management asks the user to provide a value for each variable that was not set.

Top

Parameters

Keyword	Description	Choices	Notes
QMQRY	Query management query	*Qualified object name*	Required, Positional 1
	Qualifier 1: Query management query	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
OUTPUT	Output	*****, *PRINT, *OUTFILE	Optional, Positional 2
QMFORM	Query management report form	Single values: ***SYSDFT**, *QMQRY Other values: *Qualified object name*	Optional
	Qualifier 1: Query management report form	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
OUTFILE	File to receive output	*Qualified object name*	Optional
	Qualifier 1: File to receive output	*Name*	
	Qualifier 2: Library	*Name*, *CURLIB, ***LIBL**	
OUTMBR	Output member options	*Element list*	Optional
	Element 1: Member	***FIRST**	
	Element 2: Replace or add records	***REPLACE**, *ADD	
DATETIME	Date and time	***YES**, *NO	Optional
PAGNBR	Page numbers	***YES**, *NO	Optional
RDB	Relational database	*Simple name*, ***NONE**, *CURRENT	Optional
RDBCNNMTH	Connection Method	***DUW**, *RUW	Optional

Keyword	Description	Choices	Notes
USER	User	*Name*, ***CURRENT**	Optional
PASSWORD	Password	*Character value*, *NONE	Optional
NAMING	Naming convention	***SYS**, *SQL, *SAA	Optional
ALWQRYDFN	Allow information from QRYDFN	***NO**, *YES, *ONLY	Optional
SETVAR	Set variables	Values (up to 50 repetitions): *Element list*	Optional
	Element 1: Variable name	*Character value*	
	Element 2: Variable value	*Character value*	

Top

Query management query (QMQRY)

Specifies the query management query (QMQRY) to be run.

This is a required parameter.

Qualifier 1: Query management query

name Specify the name of the query to run.

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 where the query is located.

Top

Output (OUTPUT)

Specifies whether the output from the command is shown at the requesting work station, printed with the job's spooled output, or directed to a database file.

***** The output produced by the query is formatted with the specified report form and, in interactive mode, sent to the work station that runs the command. If the command is run in batch mode, the output is sent to the default printer used by query management.

***PRINT**
The output produced by the query is formatted with the specified query management form, then sent to the default printer used by query management.

***OUTFILE**
The output produced by the query is written to a database file (table), which is inserted into a collection.

Top

Query management report form (QMFORM)

Specifies which query management report form is to be applied to the answer-set to format the printed or displayed output.

Single values

***SYSDFT**
A default report form is created and used for the report that is printed or displayed.

***QMQRY**
The value specified on the **Query management query (QMQRY)** parameter is used to locate the report form.

Qualifier 1: Query management report form

name Specify the name of the report form.

Qualifier 2: Library

***LIBL** The library list is used to locate the report form.

***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.

Top

File to receive output (OUTFILE)

Specifies the database file that receives the query output. If the file specified does not exist, the system creates it in the specified library as a table in a collection. If the file is created by this function, the authority for users without specific authority is *EXCLUDE.

Qualifier 1: File to receive output

name Specify the name of the database file that receives the output of the command.

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.

Top

Output member options (OUTMBR)

Specifies the database file member to which the output is directed.

Element 1: Member

***FIRST**
The first member in the file receives the output.

Element 2: Replace or add records

***REPLACE**
The file is cleared before new records are inserted.

***ADD** New records are added after any existing records.

Top

Date and time (DATETIME)

Specifies whether the system date and time are printed on the bottom of each page.

***YES** The system date and time are printed on the bottom of each page.

***NO** The system date and time are not printed on the bottom of each page.

Top

Page numbers (PAGNBR)

Specifies whether page numbers are printed on the bottom of each page.

***YES** The page numbers are printed on the bottom of each page.

***NO** The page numbers are not printed.

Top

Relational database (RDB)

Specifies the relational database that is accessed during the processing of this command.

***NONE**
The local database is accessed. If you are connected to a remote database, the connection is reset to local and remains local until completion of this command. If the connection management method is *DUW the remote connection is left in a dormant state.

***CURRENT**
The relational database to which you are currently connected is accessed.

With *RUW connection management, if the user is connected to a remote database, *OUTFILE cannot be specified on the OUPUT parameter.

With *DUW connection management, if the user is connected to a remote database and OUTPUT(*OUTFILE) is specified, the connection is set to local for the *OUTFILE processing and then the remote connection is restored when the STRQMQRY command is completed.

name Specify the name of the relational database that is accessed. The database you specify must have an entry in the relational database directory.

With *RUW connection management, if the relational database specified is a remote database and OUTPUT(*OUTFILE) is specified, the connection is reset to local for the *OUTFILE processing and remains local when the STRQMQRY command is completed.

With *DUW connection management, if the relational database and OUTPUT(*OUTFILE) is specified, the connection is set to local for the *OUTFILE processing and then the remote connection is restored upon completion of the STRQMQRY command.

Top

Connection Method (RDBCNNMTH)

Specifies the connection method to use.

***DUW**
Connections to several relational databases are allowed. Consecutive CONNECT statements to additional relational databases do not result in the disconnection of previous connections.

***RUW** Only one connection to a relational database is allowed. Consecutive CONNECT statements result in the previous connections being disconnected before a new connection is established.

Top

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.

Top

Password (PASSWORD)

Specifies the password to be used on the remote system.

***NONE**
No password is sent. The user name specified on the USER parameter is not valid if this value is specified.

character-value
Specify the password of the user name specified on the USER parameter.

Top

Naming convention (NAMING)

Specifies the naming convention used for naming objects.

***SYS** The system naming convention is used (database-name/object-name).

***SQL** The SQL naming convention is used (database-name.object-name). If NAMING(*SQL) is specified, the *LIBL value cannot be specified or allowed to be a default value for locating any of the objects specified on other parameters on this command.

***SAA** The SQL naming convention is used (database-name.object-name). If NAMING(*SAA) is specified, the *LIBL value cannot be specified or allowed to be a default value for locating any of the objects specified on other parameters on this command.

Top

Allow information from QRYDFN (ALWQRYDFN)

Specifies whether query or form information is taken from a query definition (QRYDFN) object when no query management query (QMQRY) or query management form (QMFORM) object can be found using the specified object name. Any information that has to be derived in this way is discarded when the command completes. No query management object is created.

***NO** Information is not taken from a QRYDFN object.

***YES** Information is taken from a QRYDFN object when the specified QMQRY or QMFORM object name is not found.

***ONLY**
Information is taken only from a QRYDFN object. Query management objects are ignored.

Top

Set variables (SETVAR)

Specifies the variables that are set by query management before the query is run. Up to 50 variables can be set.

Element 1: Variable name

character-value
Specify a variable name, from 1 to 30 characters. Because lower-case characters in variable names are changed to upper-case characters when passed to the command processing program, you cannot use this parameter to set values for variables with mixed case names.

Element 2: Variable value

character-value
Specify a variable value, from 0 to 55 characters. If you enclose a value in apostrophes, the apostrophes are removed and double apostrophes within the value are condensed to single apostrophes when the value is passed to the command processing program.

Top

Examples

Example 1: Displaying Query Output

```
STRQMQRY   QMQRY(MYLIB/MYQRY)  QMFORM(FORM1)
```

This command runs query management query MYQRY located in library MYLIB. The library list is searched for form FORM1, which is used for the output sent to the display.

Example 2: Taking Information From Either QMQRY or QRYDFN

```
STRQMQRY   QMQRY(MYLIB/MYQRY)  QMFORM(FORM1)  ALWQRYDFN(*YES)
```

This command runs query management query (QMQRY) MYQRY located in library MYLIB. If QMQRY object MYQRY is not found in library MYLIB, the information is taken from query definition (QRYDFN) MYQRY located in library MYLIB. The library list is searched for query management form FORM1 whose information is used to format the output. If QMFORM object FORM1 is not found in the library list, the library list is searched for QRYDFN FORM1, and that information is used to format the output shown on the display.

Example 3: Printing Query Output

```
STRQMQRY   QMQRY(MYLIB/QUERY1)  OUTTYPE(*PRINTER)
```

This command runs query QUERY1 located in library MYLIB. The report is formatted and printed on the printer specified in the printer file associated with the query session.

Example 4: Sending Output to an Existing File

```
STRQMQRY   QMQRY(*CURLIB/MYQRY)
           OUTPUT(*OUTFILE)  OUTFILE(MYTAB)  OUTMBR(*FIRST *ADD)
```

This command runs the query named MYQRY located in the current library for the user's job. The selected data records are added to the previously created table named MYTAB in collection MYCOL.

Example 5: Running a Query Containing Substitution Variables

```
STRQMQRY   QMQRY(MYQUERY)
           SETVAR((VAR1 'select * from mytable')
                  (VAR2 'where salary > 15000'))
```

This command runs query MYQUERY, which contains only substitution variables, &VAR1 and &VAR2. These two variables contain the entire structured query language (SQL) statement.

Example 6: Changing a Variable

```
STRQMQRY   QMQRY(QRYNAME)  SETVAR((LASTNAME '''Smith'''))
```

This command runs query QRYNAME, setting the variable LASTNAME to the value, 'Smith'.

Top

Error messages

***ESCAPE Messages**

QWM2701
&1 command failed.

QWM2703
&1 command ended.

QWM2707
*LIBL not allowed when SQL naming applied.

QWM2709
User or password not valid with relational database value.

QWM2710
Password value *NONE only valid with user value *CURRENT.

QWM2712

Character in user name not valid.

Top

Start Query (STRQRY)

Where allowed to run: Interactive environments (*INTERACT *IPGM *IREXX *EXEC)
Threadsafe: No

Parameters
Examples
Error messages

The Start Query (STRQRY) command displays the Query for i5/OS main menu.

There are no parameters for this command.

Top

Parameters

None

Top

Examples

```
STRQRY
```

This command shows the main Query Utilities menu.

Top

Error messages

None

Top

Start QSH (STRQSH)

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

Parameters
Examples
Error messages

The Start QSH (STRQSH) command, as known as QSH, starts the **qsh** shell interpreter.

If run in an interactive job, STRQSH starts an interactive shell session. If a shell session is not already active in the job, then:

1. A new shell session is started and a terminal window is displayed.
2. **qsh** runs the commands from the file /etc/profile if it exists.
3. **qsh** runs the commands from the file .profile in the user's home directory if it exists.
4. **qsh** runs the commands from the file specified by the expansion of the ENV variable if it exists.

If a shell session is already active in an interactive job, you are reconnected to the existing session. From the terminal window, you can enter shell commands and view output from the commands.

Using the Terminal Window

The terminal window has two parts:

- an input line for entering commands, and
- an output area that contains an echo of the commands you entered and any output generated by the commands.

The terminal window supports the following function keys:

F3 (Exit)
Close the terminal window and end the qsh session.

F5 (Refresh)
Refresh the output area.

F6 (Print)
Print the output area to a spool file.

F7 (Up)
Roll output area up one page.

F8 (Down)
Roll output area down one page.

F9 (Retrieve)
Retrieve a previous command. You can press this key multiple times to retrieve any previous command. For example, to retrieve the second to last command you entered, press this key two times. You can also select a specific command to be run again by placing the cursor on that command and pressing this key. When the interactive job is running in a double-byte CCSID, this key is not available.

F11 (Toggle line wrap)
Toggle the line wrap/truncate mode in the output area. In line wrap mode, lines longer than the width of the terminal window are wrapped to the next line. In truncate mode, the portion of a line beyond the width of the terminal window is not shown.

F12 (Disconnect)
Disconnect from the **qsh** session. This key only closes the terminal window and does not end the **qsh** session. You can redisplay the disconnected **qsh** session by running STRQSH again.

F13 (Clear)
Clear the output area.

F17 (Top)
Display top of output area.

F18 (Bottom)
Display bottom of output area.

F19 (Left)
Shift output area to the left.

F20 (Right)
Shift output area to the right.

F21 (CL command entry)
Display a command entry window where you can enter CL commands.

Also, you can use SysReq 2 to interrupt the currently running command.

Error messages for STRQSH

***ESCAPE Messages**

QSH0002
Error found with QSH session, reason code &1, errno &2.

Top

Parameters

Keyword	**Description**	**Choices**	**Notes**
CMD	Command	*Character value*, ***NONE**	Optional

Top

Command (CMD)

Specifies the shell command to be run.

The possible values are:

***NONE:**
No command is provided and an interactive session is started.

command
A shell command to run. The command can be a maximum of 5000 bytes in length. If a blank or other special characters are used, the command must be enclosed in apostrophes (′). If an apostrophe is intended, two apostrophes must be used (′′).

Note: The case is preserved when lowercase characters are specified.

Top

Examples

None

Top

Error messages

*ESCAPE Messages

QSH0002

Error found with QSH session, reason code &1, errno &2.

Top

Start Question and Answer (STRQST)

Where allowed to run: Interactive environments (*INTERACT *IPGM *IREXX *EXEC)
Threadsafe: No

Parameters
Examples
Error messages

The Start Question and Answer (STRQST) command shows the main Question & Answer (Q & A) menu. More information is available in the Basic system operations topic collection in the i5/OS Information Center at http://www.ibm.com/systems/i/infocenter/.

Top

Parameters

Keyword	Description	Choices	Notes
QSTDB	Q/A database	*Name*, ***SELECT**	Optional, Positional 1
LIB	Lib containing Q/A database	*Name*, ***QSTLIB**	Optional, Positional 2

Top

Q/A database (QSTDB)

Specifies the Question and Answer (Q & A) database with which to work.

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 with which to work.

Top

Lib containing Q/A database (LIB)

Specifies the name of the library that contains the Q & A database.

The name of the Q & A database can be qualified by one of the following library values:

***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.

Top

Examples

```
STRQST
```

This command shows the Question and Answer (Q & A) main menu.

Top

Error messages

None

Top

Start REXX Procedure (STRREXPRC)

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

Parameters
Examples
Error messages

The Start REXX Procedure (STRREXPRC) command calls the interpreter, explicitly specifying the library, file, and source member to interpret.

Top

Parameters

Keyword	Description	Choices	Notes
SRCMBR	Source member	*Name*	Required, Positional 1
SRCFILE	Source file	*Qualified object name*	Optional
	Qualifier 1: Source file	*Name*, **QREXSRC**	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
PARM	Parameters	*Character value*, ***NONE**	Optional
CMDENV	Command environment	Single values: ***COMMAND**, *CPICOMM, *EXECSQL Other values: *Qualified object name*	Optional
	Qualifier 1: Command environment	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
EXITPGM	Exit program	Single values: ***NONE** Other values (up to 8 repetitions): *Element list*	Optional
	Element 1: Program	*Qualified object name*	
	Qualifier 1: Program	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
	Element 2: Exit code	2, 3, 4, 5, 7, 8, 9, 10	

Top

Source member (SRCMBR)

Specifies the name of the source file member that contains the REXX procedure to be started.

This is a required parameter.

Top

Source file (SRCFILE)

Specifies the source file that contains the REXX procedure to be started.

Qualifier 1: Source file

QREXSRC
The IBM-supplied source file, QREXSRC, contains the REXX procedure.

name Specify the name of the source file that contains the REXX procedure to be run.

Qualifier 2: Library

***LIBL** The library list is used to locate the source file.

When *LIBL is specified for this qualifier, REXX will find the first source file in the library list with the specified source file name. If a source file is found, REXX will look for the source member in only that source file. REXX will not continue looking for any other source files by the same name in the library list. This is the default REXX source file member search behavior.

You can change how REXX searches for the source file member by adding a job-level environment variable named QIBM_REX_FIND_SRCMBR. You would use the Add Environment Variable (ADDENVVAR) command and specify ENVVAR(QIBM_REX_FIND_SRCMBR) and LEVEL(*JOB).

- If the initial value (VALUE parameter) specified on the ADDENVVAR command is *BYMBRNAME, REXX will search for the source member by looking in each source file with the specified file name in the library list that contains a source member with the name specified for the **Source member (SRCMBR)** parameter.
- If the initial value specified on the ADDENVVAR command is *BYFILENAME or any other value, or if the environment variable does not exist, the default REXX source file member search behavior is used.

***CURLIB**
The current library 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.

Top

Parameters (PARM)

Specifies procedure parameter values passed to the REXX procedure when it is started. These values are accessed through the argument (ARG) instruction within the REXX procedure.

***NONE**
There are no procedure parameters for the REXX procedure. The ARG instruction returns a null string.

character-value
Specify the procedure parameter value to pass to the REXX procedure. A maximum of 3000 characters is allowed.

Top

Command environment (CMDENV)

Specifies the initial command environment program to process commands embedded in the REXX procedure. The REXX interpreter will call this environment whenever a command is encountered within the REXX procedure.

Single values

***COMMAND**
The System i control language (CL) command environment is used.

***CPICOMM**
The Common Programming Interface (CPI) for Communications command environment is used.

***EXECSQL**
The Structured Query Language (SQL) Command environment is used. EXECSQL is the command environment used for CL commands that are imbedded within a REXX procedure.

Qualifier 1: Command environment

name Specify the name of the program to process commands embedded in the REXX procedure.

Qualifier 2: Library

***LIBL** The library list is used to locate the program.

***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 program is located.

Top

Exit program (EXITPGM)

Specifies exit programs to be used when the interpreter is called. A maximum of 8 program and exit code pairs can be specified.

Note: If multiple system exit codes are specified, the last one specified is the one taken.

Single values

***NONE**
There are no exit programs for this call.

Other values (up to 8 repetitions)

Element 1: Program

Qualifier 1: Program

name Specify the name of the exit program.

Qualifier 2: Library

***LIBL** The library list is used to locate the specified program.

***CURLIB**
The current library for the job is used to locate the specified 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 specified program is located.

Element 2: Exit code

number
Specify one of the following exit code values.

Exit-code
Description

2 The associated program is called whenever an external function or subroutine has been called by the REXX program. The exit program is then responsible for locating and calling the requested routine.

3 The associated program is called whenever the interpreter is going to call a command. The exit program is responsible for locating and calling the command given the command string and the current environment name.

4 The associated program is called whenever a REXX instruction or function attempts an operation on the REXX external data queue.

5 The associated program is called when session input or output operations are attempted.

7 The associated program is called after running each clause of the REXX procedure to determine whether it should be stopped.

8 The associated program is called after running each clause of the REXX program to check if tracing should be turned on or off.

9 The associated program is called before interpretation of the first instruction of a REXX procedure (including REXX procedures called as external functions and subroutines).

10 The associated program is called after interpretation of the last instruction of a REXX procedure (including REXX procedures called as external functions and subroutines).

Top

Examples

```
STRREXPRC   SRCMBR(ABC)
```

This command calls the REXX interpreter instructing it to run the source member named ABC in the first QREXSRC source file in the library list (*LIBL).

Top

Error messages

***ESCAPE Messages**

CPF7CFB
Error occurred while processing REXX exit programs.

CPF7CFD
Error occurred running REXX procedure &1.

CPF7CFF
REXX procedure &1 ended; return code &4.

CPF7CF2
REXX procedure &1 not found.

CPF7CF3
Not authorized to source file &2.

CPF7CF4
Cannot allocate REXX procedure &1.

CPF7CF6
Cannot allocate REXX source file &2.

CPF7CF7
REXX external data queue is damaged.

Top

Start Remote Support (STRRMTSPT)

Where allowed to run: Interactive environments (*INTERACT *IPGM *IREXX *EXEC)
Threadsafe: No

Parameters
Examples
Error messages

The Start Remote Support (STRRMTSPT) command creates and varies on all configuration objects needed for remote support. Remote support allows the IBM service organization to access your system. Remote support options that are available include:

- remote work station
- virtual device
- virtual device over a systems network architecture (SNA) connection
- virtual device over a point-to-point protocol (PPP) connection using internet protocol (IP)
- virtual device over a virtual private network (VPN) connection using internet protocol (IP)

Each remote support option has different command parameters that are required for connectivity. If any existing remote support configuration objects are found, they are deleted and then re-created. After the configuration objects have been created, they are varied on. You must provide a user identifier and password before the support person can sign-on your system.

Restrictions:

1. When using DEVCLS(*RMT), the remote work station used by a support organization must be one of those listed in the DSPTYPE and DSPMODEL parameters. If your support organization has a configuration that does not match, you must work with the support person to create the correct configuration objects on your system.
2. When using DEVCLS(*VRT) or DEVCLS(*IPS), the QUSER user profile must not be disabled.

Top

Parameters

Keyword	Description	Choices	Notes
DEVCLS	Device class	*RMT, *VRT, *IPS, *PPP, *VPN	Required, Positional 1
DSPTYPE	Display type	3179, 3180, 3196, 3197, 5251, 5291, 5292	Optional, Positional 2
DSPMODEL	Display model	1, 2, 11, A1, A2, B1, B2, C1, C2, D1, D2, W1, W2, 0001, 0002, 0011	Optional, Positional 3
STNADR	Station address	01-FE, **FE**	Optional, Positional 4
USRPRF	User profile	*Character value*, **QPGMR**	Optional, Positional 5
RSRCNAME	Resource name	*Character value*, ***DFT**	Optional, Positional 6
RMTLOCNAME	Remote location	*Communications name*, **QREMOTE**	Optional, Positional 7
LCLLOCNAME	Local location	*Communications name*, **QLOCAL**, *NETATR	Optional, Positional 8
RMTNETID	Remote network identifier	*Communications name*, *NETATR, ***NONE**	Optional

Keyword	Description	Choices	Notes
MAXLENRU	Maximum length of request unit	241-32767, **32767**, *SAME, *CALC	Optional
DTACPR	Data compression	*SAME, *NETATR, *NONE, *ALLOW, ***REQUEST**, *REQUIRE	Optional
INDTACPR	Inbound data compression	*SAME, *RLE, *LZ9, *LZ10, ***LZ12**, *NONE	Optional
OUTDTACPR	Outbound data compression	*SAME, *RLE, *LZ9, *LZ10, ***LZ12**, *NONE	Optional
MODEM	Modem	*Character value*, ***RSRCNAME**, *RMTPPP, *SELECT	Optional
MDMRMTSYS	Modem remote system	*Character value*	Optional

Top

Device class (DEVCLS)

Specifies the device class for this display station. Different configuration objects are created, depending on the value specified.

This is a required parameter.

***RMT** This device class is for a device connected to a remote work station. The configuration objects created for this option include a line description, a controller description, a display device description, and a printer device description.

***VRT** This device class is for a virtual device. The configuration objects created for this option include a line description, a controller description, a device description, a virtual control description, and a virtual display device description.

***IPS** This device class is for a virtual device and also supports internet protocol (IP) over SNA sessions on this connection. The configuration objects created for this option include a line description, a controller description, a device description, a virtual control description, and a virtual display device description.

***PPP** This device class supports IP over a point-to-point protocol (PPP) connection. The configuration objects created for this option include a line description, a controller description, a device description, and a PPP profile. DEVCLS(*PPP) requires that the QRETSVRSEC system value be set to ’1’ to retain server security data. This allows for additional authentication to be performed when the service organization attempts to connect.

***VPN** This device class supports IP over a virtual private network (VPN) connection. The Universal Connection wizard, or the Create Service Configuration (CRTSRVCFG) command must be run before running the STRRMTSPT command in order to set up a VPN configuration on the system. No configuration objects are created for this option.

Top

Display type (DSPTYPE)

Specifies the workstation display device type which will be used for remote support. A value must be specified for this parameter when DEVCLS has a value of *RMT, *VRT, or *IPS.

Valid display station type values are:

- 3179
- 3180
- 3196
- 3197

- 5251
- 5291
- 5292

Top

Display model (DSPMODEL)

Specifies the model number of the device for this description. A value must be specified for this parameter when DEVCLS has a value of *RMT, *VRT, or *IPS.

The possible values for the device model for each device type are:

TYPE	**MODEL**
3179	2
3180	2
3196	A1, A2, B1, or B2
3197	C1, C2, D1, D2, W1, or W2
5251	11
5291	1, or 2
5292	1, or 2

Top

Address (STNADR)

Depending on the value specified for the DEVCLS parameter, this parameter specifies the address that will be used for either the station address or the local IP address qualifier. If the DEVCLS is not *PPP or *VPN, then this parameter specifies the station address. The station address is the hexadecimal address by which the local system is known to the remote system. The hexadecimal address is the polling address assigned to this system.

If the DEVCLS is *IPS, *PPP, or *VPN, this parameter is used as the local IP address qualifier. This specifies the value of the last byte of the local internet address that will be used.

FE The hexadecimal value FE is the local system address.

address
Specify a hexadecimal value from 01 to FE.

Top

User profile (USRPRF)

Specifies the name of the user profile that the IBM service personnel will use to sign on to your system. This profile is made the owner of the objects created by the system when remote support is started. The user profile must already exist on your system.

QPGMR
The default system-supplied user profile, QPGMR, is used to sign on to your system.

user-profile-name
Specify the name of the existing user profile that will be used to sign on to the remote system.

Top

Resource name (RSRCNAME)

Specifies the name of the resource used to access electronic customer support. This parameter is not used for DEVCLS(*VPN).

***DFT**

- For DEVCLS(*PPP):
  - Look for resources being used by the 2771 or 2793 integrated modem. If only one 2771 or 2793 is defined, that resource is used for this PPP line. Note that more than one 2771 or 2793 could be defined, but a 2771 or 2793 resource can only be calculated if only one is defined.
  - If a 2771 or 2793 modem resource cannot be used, determine if any resources are defined for use by electronic customer support (ECS). If an ECS resource is available, that resource is used for this PPP line.
  - If neither a single 2771 or 2793 integrated modem or ECS resource is available, the resource cannot be calculated and it will have to be explicitly specified.
- For all other DEVCLS values (except *VPN): The resource name associated with the shipped default port for accessing electronic customer support is used. This is the first port on the I/O (input/output) adapter in card position B of the first multifunction IOP (input/output processor) on the bus. If this port does not exist on the system, resource name CMN01 is used.

resource-name
Specify the name of the resource used to access electronic customer support.

Top

Remote location (RMTLOCNAME)

Specifies the remote location name of the system with which this object communicates. This parameter is used when DEVCLS is *VRT or *IPS.

QREMOTE
The default system-supplied remote support location name.

remote-location-name
Specify the name of the remote support location.

Top

Local location (LCLLOCNAME)

Specifies the unique location name that identifies the local system to remote devices. The name cannot be the same as that specified for the **Remote location** prompt (RMTLOCNAME parameter). The combination of the names specified for the **Local location** prompt (LCLLOCNAME parameter) and the **Remote location** prompt (RMTLOCNAME parameter) must be unique for each device attached to the same controller. This parameter is used when the DEVCLS parameter has a value of *VRT or *IPS.

QLOCAL
The default system-supplied local location name is used.

***NETATR**
The LCLLOCNAME value specified in the system network attributes is used.

local-location-name
Specify the location name to be used to identify the local system to remote devices.

Top

Remote network identifier (RMTNETID)

Specifies the name of the remote network in which the remote system resides. This parameter is used when the DEVCLS parameter has a value of *VRT or *IPS.

***NONE**
No remote network name is used.

remote-network-name
Specify the remote network name.

Top

Maximum length of request unit (MAXLENRU)

Specifies the maximum request unit (RU) length allowed. This parameter is used when the DEVCLS parameter has a value of *VRT or *IPS.

32767 The maximum request unit length is 32767 bytes.

***CALC**
The system calculates the maximum value to use.

maximum-request-unit-length
Specify a value, ranging from 241 through 32767 bytes, for the maximum length of incoming request units.

Some common values, based on line type, are:

- SDLC lines: 256, 512, 1024, 2048
- Token-Ring Network lines: 256, 512, 1024, 1985
- X.25 (QLLC) lines: 247, 503, 1015
- X.25 (ELLC) lines: 241, 497, 1009

More information on setting the maximum RU length is in the Communications Configuration book, SC41-5401 book.

Top

Data compression (DTACPR)

Specifies whether data compression is used.

***REQUEST**
Data compression is requested on the session by the local system. However, the request can be refused or changed to lower compression levels by the remote system. Data compression is allowed on the session if requested by the remote system. The requested compression levels for inbound and outbound data are the levels specified on the **Inbound data compression** and **Outbound data compression** prompts (INDTACPR and OUTDTACPR parameters).

If data compression is requested by the remote system, the data compression levels used by the session are the lower of the requested levels and the levels specified on the **Inbound data compression** and **Outbound data compression** prompts (INDTACPR and OUTDTACPR parameters).

***NONE**
Compression is not allowed on the session.

***ALLOW**
Data compression is allowed on the session by the local system if requested by a remote system. The local system does not request compression.

If data compression is requested by the remote system, the data compression levels used by the session are the lower of the requested levels and the levels specified on the **Inbound data compression** and **Outbound data compression** prompts (INDTACPR and OUTDTACPR parameters).

***REQUIRE**
Data compression is required on the session. If the remote system does not accept the local system's exact required levels of compression, the session is not established.

The data compression levels that the local system require are the levels specified on the **Inbound data compression** and **Outbound data compression** prompts (INDTACPR and OUTDTACPR parameters).

Top

Inbound data compression (INDTACPR)

Specifies the desired level of compression for inbound data. No data compression occurs if *NONE is specified on the **Data compression** prompt (DTACPR parameter). Adaptive dictionary-based compression is a dynamic compression algorithm, similar to Lempel-Ziv, that compresses previously seen strings to 9-, 10-, and 12-bit codes. This algorithm is referred to as LZ in the following parameters.

***LZ12** The LZ algorithm with the 12-bit code for repeated substrings in the data stream is used. These codes refer to entries in a common dictionary, created as the data flows between the sender and receiver. The LZ algorithms require storage and extra processing time. LZ12 requires the most storage and processing time of the LZ algorithms; however, it compresses the data stream the most.

***RLE** The Run Length Encoding (RLE) algorithm is used. RLE substitutes a 1- or 2-byte sequence in the data stream for each repeated occurrence of the same character. This algorithm requires no storage and less processing time than the other options.

***LZ9** The LZ algorithm with the 9-bit code for repeated substrings in the data stream is used. These codes refer to entries in a common dictionary, created as the data flows between the sender and receiver. The LZ algorithms require storage and extra processing time. The LZ9 requires the least storage and processing time of the LZ algorithms; however, it compresses the data stream the least.

***LZ10** The LZ algorithm with the 10-bit code for repeated substrings in the data stream is used. These codes refer to entries in a common dictionary, created as the data flows between the sender and receiver. The LZ algorithms require storage and extra processing time. The LZ10 table algorithm requires more storage and processing time than the LZ9, but less than the LZ12. The LZ10 compresses the data stream more than the LZ9, but less than the LZ12.

Top

Outbound data compression (OUTDTACPR)

Specifies the desired level of compression for outbound data. No data compression occurs if *NONE is specified on the **Data compression** prompt (DTACPR parameter).

***LZ12** The LZ algorithm with the 12-bit code for repeated substrings in the data stream is used. These codes refer to entries in a common dictionary, created as the data flows between the sender and receiver. The LZ algorithms require storage and extra processing time. LZ12 requires the most storage and processing time of the LZ algorithms; however, it compresses the data stream the most.

***RLE** The Run Length Encoding (RLE) algorithm is used. RLE substitutes a 1- or 2-byte sequence in the data stream for each repeated occurrence of the same character. This algorithm requires no storage and less processing time than the other options.

***LZ9** The LZ algorithm with the 9-bit code for repeated substrings in the data stream is used. These codes refer to entries in a common dictionary, created as the data flows between the sender and receiver. The LZ algorithms require storage and extra processing time. The LZ9 requires the least storage and processing time of the LZ algorithms; however, it compresses the data stream the least.

***LZ10** The LZ algorithm with the 10-bit code for repeated substrings in the data stream is used. These codes refer to entries in a common dictionary, created as the data flows between the sender and receiver. The LZ algorithms require storage and extra processing time. The LZ10 table algorithm requires more storage and processing time than the LZ9, but less than the LZ12. The LZ10 compresses the data stream more than the LZ9, but less than the LZ12.

Top

Modem (MODEM)

Specifies the modem description to use for the point-to-point protocol (PPP) profile.

***RSRCNAME**

The modem description will be determined based on the value defined for the RSRCNAME parameter.

- If the resource is defined to use the 2771 integrated modem, the ’2771 Internal Modem’ description is used.
- If the resource is defined to use the 2772 integrated modem, the ’2772 Internal Modem’ description is used.
- If the resource is defined to use the 2793 integrated modem, the ’2793 Internal Modem’ description is used.
- If the resource is defined to use the 2805 integrated modem, the ’2805 Internal Modem’ description is used.
- If the ECS resource was chosen, the ’IBM 7852-400’ modem description is used.
- Otherwise if the resource does not have a pre-defined modem description, MODEM(*RSRCNAME) cannot be used and the modem description will have to be explicitly defined.

***RMTPPP**

Indicates that a modem on a different partition or system is to be used. The internet address or host name where the modem is located must be specified for the **Modem remote system (MDMRMTSYS)** parameter.

***SELECT**

A list of modem descriptions is shown from which you can select the modem to use. This option is only valid when running the STRRMTSPT CL command in interactive mode, otherwise an error will occur. If you are running interactively, it is recommended that you use the *SELECT value to help ensure that you properly select the modem to use.

’modem-identification’

Specify the name of the modem to use. Note that modem names are case sensitive and must match exactly to the modems defined for the system.

'generic-modem-identification'*

Specify the generic name of the modem you wish to use. A generic modem name is a character string of one or more characters followed by an asterisk (*); for example, 'abc*'. If a generic name is specified, then the FIRST modem name that matches with the generic name will be used. It is recommended that you include as many characters in the modem name string as possible to avoid any ambiguity. If an asterisk is not included with the generic (prefix) name, the system assumes it to be the complete modem name. The actual modem name chosen will be posted in a message in the joblog.

Top

Modem remote system (MDMRMTSYS)

Specifies the internet protocol (IPv4) address or the host name for the system or partition where the modem to be used for remote support is located.

Note: This parameter is required if *RMTPPP is specified for the **Modem (MODEM)** parameter. If the MODEM parameter has any value other than *RMTPPP, this parameter cannot be specified.

character-value

Specify the IP address or host name of the remote system or partition where the modem is located.

Top

Examples

Example 1: Start Remote Support using PPP

```
STRRMTSPT   DEVCLS(*PPP)  STNADR(FA)
```

This command creates and starts a PPP answer profile and associated configuration objects. The local internet address assigned is 169.254.2.250.

Example 2: Start Remote Support for Virtual Device

```
STRRMTSPT   DEVCLS(*VRT)  DSPTYPE(5251)
            DSPMODEL(11)  STNADR(FE)
```

This command creates and varies on the 5251 Model 11 Display Station located at station address FE.

Example 3: Start Remote Support using VPN

```
STRRMTSPT   DEVCLS(*VPN)
```

This command starts remote support over a virtual private network (VPN) connection. Before running this command, the Universal Connection wizard, or the Create Service Configuration (CRTSRVCFG) command needs to be run to create the VPN configuration.

Example 4: Start Remote Support using a Remote Modem and PPP

```
STRRMTSPT   DEVCLS(*PPP)  STNADR(FE)  MODEM(*RMTPPP)
            MDMRMTSYS('10.1.1.2')
```

This command creates and starts an L2TP remote answer profile and associated configuration objects. Before running this command, the Universal Connection wizard needs to be run on the system or partition with internet address 10.1.1.2 where the modem to be used is located. When running the Universal Connection wizard, you need to specify that the modem should provide connectivity to others and that the connection type is AT&T. An alternative would be to create an L2TP terminator profile using the Remote Access Service New Profile GUI on the system or partition where the modem is located, specifying that outgoing calls are supported.

Top

Error messages

***ESCAPE Messages**

CPF1394
CPF1394 User profile &1 cannot sign on.

CPF9801
Object &2 in library &3 not found.

CPF9899
Error occurred during processing of command.

TCP83A7
QRETSVRSEC=0. Unable to save remote service password for &2.

Top

Start Remote Writer (STRRMTWTR)

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

Parameters
Examples
Error messages

The Start Remote Writer (STRRMTWTR) command starts a spooling writer that sends spooled files from an output queue to a remote system. The writer, which is a system job, takes spooled files from an output queue and sends them to a remote system using SNADS or TCP/IP.

After the spooled file is successfully sent to a remote system, the spooled file will be deleted or saved, as specified by the SAVE spooled file attribute.

More than one writer can be active at the same time (as determined by the spooling subsystem description), and up to 10 writers can be active to the same output queue. Each writer must have a unique writer name, and only one type (print, remote, or diskette) of writer can be active to a single output queue. A writer that has been started can be actively sending output or waiting for a file to be put on the output queue. Optionally, the writer can end automatically when it has processed all the files on the output queue. You can also change, hold, or cancel the writer.

You can continue with other work after starting a writer because each job runs independently. The writer is owned by the user who issues the STRRMTWTR command.

Top

Parameters

Keyword	Description	Choices	Notes
OUTQ	Output queue	Single values: *ALL Other values: *Qualified object name*	Required, Positional 1
	Qualifier 1: Output queue	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
MSGQ	Queue for writer messages	Single values: ***OUTQ**, *REQUESTER Other values: *Qualified object name*	Optional, Positional 3
	Qualifier 1: Queue for writer messages	*Name*	
	Qualifier 2: Library	*Name*, ***LIBL**, *CURLIB	
FORMTYPE	Form type options	*Element list*	Optional
	Element 1: Form type	*Character value*, ***ALL**, *STD, *FORMS	
	Element 2: Message option	***NOMSG**, *INQMSG, *MSG, *INFOMSG	
WTR	Writer	*Name*, ***OUTQ**, *SYSGEN	Optional, Positional 2
AUTOEND	Auto-end options	*Element list*	Optional, Positional 4
	Element 1: Automatically end writer	***NO**, *YES	
	Element 2: If yes, when to end	***NORDYF**, *FILEEND	

Top

Output queue (OUTQ)

Specifies the output queue that holds the spooled files to be written.

Single values

***ALL** Starts remote writers for every output queue on the system which have a remote system specified. No writer starts if an output queue has a different type of writer already started. The number of ″writers to autostart″ attribute in the output queue determines the number of writers started to a particular queue. This will determine how many remote writer jobs will be sending spooled output to a remote system, from a single output queue.

Qualifier 1: Output queue

name Specify the name of the output queue from which the writer processes output 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.

Top

Queue for writer messages (MSGQ)

Specifies the message queue to which messages created by the writer are sent.

Single values

***OUTQ**
Messages are sent to the message queue of the output queue specified on the OUTQ parameter.

***REQUESTER**
The messages are sent to the workstation message queue of the workstation of the user who started the writer. If this value is specified for a batch job, *OUTQ is used.

Qualifier 1: Queue for writer messages

name Specify the name of the message queue to which messages created by the 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.

Top

Form type options (FORMTYPE)

Specifies the name of the form type. The writer uses this value to select spooled files from the specified output queue for sending to a remote system.

Note: A spooled file’s form type is specified in the device file that produced the spooled file.

Element 1: Form type

***ALL** All form types are processed by the writer.

***FORMS**
The writer first chooses the first available spooled file on the output queue. After the first spooled file is complete, all spooled files with the same form type are processed. The writer then chooses the first available spooled file on the output queue and repeats the process for that form type.

***STD** Only spooled files that specify the standard form type are selected.

form-type
Specify the form type of the spooled files being produced.

Element 2: Message option

***NOMSG**
Neither an inquiry message nor an informational message is sent to the message queue.

***INQMSG**
An inquiry message is sent to the message queue when a spooled file has a form type that is different than the form type last sent.

***INFOMSG**
An informational message is sent to the message queue when no spooled files requiring this form type remain in the output queue.

***MSG** An inquiry message is sent to the message queue when a spooled file has a form type that is different than the form type in the remote and an informational message is sent when no spooled files requiring this form type remain in the output queue.

Top

Writer (WTR)

Specifies the spooling writer being started. Each writer name must be unique.

***OUTQ**
The name of the writer is the same as that of the output queue specified on the OUTQ parameter. If OUTQ(*ALL) is specified, and you have more than one writer to start, the name for each additional writer is taken from the first nine characters of the output queue followed by a digit. This last digit is sequentially assigned, starting with 2 for the second writer, 3 for the third writer, and continuing through 0 for the tenth writer.

***SYSGEN**
The writer name is generated by the system, starting with ’RMTW000001’ and incrementing the numeric part for each successive writer.

name Specify the name by which the writer being started is identified.

Top

Auto-end options (AUTOEND)

Specifies whether the writer ends automatically.

Element 1: Stop Writer Option

<u>*NO</u> The writer does not end when the last available file has been removed from the output queue. It waits for another spooled file entry to be put on the queue. This is a single value (Element 2 is not specified).

***YES** The writer automatically ends after it has reached the condition specified on the second element of this parameter.

Element 2: Conditions for Stopping Writer

<u>*NORDYF</u>
The writer automatically ends when there are no ready files (all the available files have been removed from the output queue).

***FILEEND**
The writer ends after it finishes processing one spooled file.

Top

Examples

```
STRRMTWTR   OUTQ(RMTOUTQ)  WTR(TOM)
```

This command starts a spooling writer named TOM. This writer takes the output from the output queue named RMTOUTQ and sends the output to the remote system and printer queue specified in the output queue RMTOUTQ. Writer messages are sent to the system operator's message queue, and the writer waits for more output when the queue is emptied.

Top

Error messages

<u>*ESCAPE Messages</u>

CPF330A
Output queue &1 has RMTSYS specified as *NONE.

CPF3305
Output queue &1 in library &2 assigned to another writer.

CPF3310
Writer &1 already started.

CPF3357
Output queue &1 in library &2 not found.

CPF3362
Objects in QTEMP not valid for parameter values.

CPF3363
Message queue &1 in library &2 not found.

Top

Start S/36 Session (STRS36)

Where allowed to run: Interactive environments (*INTERACT *IPGM *IREXX *EXEC)
Threadsafe: No

Parameters
Examples
Error messages

The Start System/36 (STRS36) command starts a System/36 environment session (if one is not active already). Even if the System/36 environment is active, this command allows the user to show a menu or run a program or procedure before showing a menu.

When the command ends, the System/36 environment returns to the active or inactive state from which this command is issued.

Restrictions: This command cannot be used if a System/36 procedure is already in process. This command cannot be placed in a procedure or in a program that is started by a procedure.

Top

Parameters

Keyword	Description	Choices	Notes
MENU	Menu	*Name*, ***SAME**	Optional, Positional 1
CURLIB	Current library	*Name*, ***SAME**	Optional, Positional 2
PRC	Procedure or program	*Name*, ***NONE**	Optional, Positional 3
FRCMNU	Force menu	***NO**, *YES	Optional

Top

Menu (MENU)

Specifies the first menu that is shown when the System/36 environment is started.

The possible values are:

***SAME**
The menu specified in the job does not change. If no menu is specified in the job, the initial menu specified in the user profile is shown.

menu-name
Specify the name of the first menu shown when the System/36 environment is started.

Top

Current library (CURLIB)

Specifies the current library to use in the System/36 environment.

The possible values are:

***SAME**
The current library does not change. If the current library is *CRTDFT and *SAME is specified, the current library is set to #LIBRARY.

library-name
Specify the name of the library you want to use as the current library in the System/36 environment.

Top

Procedure or program (PRC)

Specifies the name of the procedure or program to run before the menu is shown.

The possible values are:

***NONE**
No procedure or program runs.

procedure-name
Specify the name of the procedure or program to run.

Top

Force menu (FRCMNU)

Specifies whether a menu is shown and what this command does in an active System/36 environment.

The possible values are:

***NO** The specified menu is not shown if the System/36 environment is active when this command is issued. This command does nothing, and the user is returned to the point at which the command was issued.

***YES** The specified menu is shown even if the System/36 environment is active when this command is issued. The current library is set and the program or procedure is run as specified in this command.

Top

Examples

```
STRS36   MENU(USER)  CURLIB(MYLIB)  PRC(INITPROC)
```

This command starts a System/36 environment session. This command:

- Displays the USER menu
- Changes the current library to MYLIB
- Runs procedure INITPROC before showing the USER menu

Top

Error messages

***ESCAPE Messages**

CPF3709
Tape devices do not support same densities.

CPF3738
Device &1 used for save or restore is damaged.

CPF3767
Device &1 not found.

Top

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
North Castle Drive
Armonk, NY 10504-1785
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 106-0032, 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

Software Interoperability Coordinator, Department YBWA
3605 Highway 52 N
Rochester, MN 55901
U.S.A.

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 document and all licensed material available for it are provided by IBM under terms of the IBM Customer Agreement, IBM International Program License Agreement, IBM License Agreement for Machine Code, or any equivalent agreement between us.

Any performance data contained herein was determined in a controlled environment. Therefore, the results obtained in other operating environments may vary significantly. Some measurements may have been made on development-level systems and there is no guarantee that these measurements will be the same on generally available systems. Furthermore, some measurements may have been estimated through extrapolation. Actual results may vary. Users of this document should verify the applicable data for their specific environment.

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.

All statements regarding IBM's future direction or intent are subject to change or withdrawal without notice, and represent goals and objectives only.

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.

If you are viewing this information softcopy, the photographs and color illustrations may not appear.

Programming interface information

This i5/OS commands publication documents intended Programming Interfaces that allow the customer to write programs to obtain the services of IBM i5/OS.

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
Distributed Relational Database Architecture
Domino
DRDA
IBM
Infoprint
InfoWindow
i5/OS
iSeries
Integrated Language Environment
Lotus
LPDA
OfficeVision
Print Services Facility
RPG/400
System i
System x
SystemView
System/36
TCS
Tivoli
WebSphere
z/OS

Adobe, the Adobe logo, PostScript, and the PostScript logo are either registered trademarks or trademarks of Adobe Systems Incorporated in the United States, and/or other countries.

Intel, Intel logo, Intel Inside, Intel Inside logo, Intel Centrino, Intel Centrino logo, Celeron, Intel Xeon, Intel SpeedStep, Itanium, and Pentium are are trademarks or registered trademarks of Intel Corporation or its subsidiaries in the United States and other countries.

Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.

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, or 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, NON-INFRINGEMENT, AND FITNESS FOR A PARTICULAR PURPOSE.

Code license and disclaimer information

IBM grants you a nonexclusive copyright license to use all programming code examples from which you can generate similar function tailored to your own specific needs.

SUBJECT TO ANY STATUTORY WARRANTIES WHICH CANNOT BE EXCLUDED, IBM, ITS PROGRAM DEVELOPERS AND SUPPLIERS MAKE NO WARRANTIES OR CONDITIONS EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OR CONDITIONS OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT, REGARDING THE PROGRAM OR TECHNICAL SUPPORT, IF ANY.

UNDER NO CIRCUMSTANCES IS IBM, ITS PROGRAM DEVELOPERS OR SUPPLIERS LIABLE FOR ANY OF THE FOLLOWING, EVEN IF INFORMED OF THEIR POSSIBILITY:

1. LOSS OF, OR DAMAGE TO, DATA;
2. DIRECT, SPECIAL, INCIDENTAL, OR INDIRECT DAMAGES, OR FOR ANY ECONOMIC CONSEQUENTIAL DAMAGES; OR
3. LOST PROFITS, BUSINESS, REVENUE, GOODWILL, OR ANTICIPATED SAVINGS.

SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF DIRECT, INCIDENTAL, OR CONSEQUENTIAL DAMAGES, SO SOME OR ALL OF THE ABOVE LIMITATIONS OR EXCLUSIONS MAY NOT APPLY TO YOU.

Printed in USA