CA XCOM™ Data Transport® for z/OS

Release Notes

Release 11.6

CA Technologies Product References

This document references the following CA Technologies products:

- CA ACF2™ (CA ACF2)
- CA Top Secret® (CA Top Secret)
- CA XCOM™ Data Transport® (CA XCOM Data Transport)
- CA XCOM™ Data Transport® Management Center (CA XCOM Management Center)

Contact CA Technologies

Contact CA Support

For your convenience, CA Technologies provides one site where you can access the information that you need for your Home Office, Small Business, and Enterprise CA Technologies products. At http://ca.com/support, you can access the following resources:

- Online and telephone contact information for technical assistance and customer services
- Information about user communities and forums
- Product and documentation downloads
- CA Support policies and guidelines
- Other helpful resources appropriate for your product

Providing Feedback About Product Documentation

If you have comments or questions about CA Technologies product documentation, you can send a message to techpubs@ca.com.

To provide feedback about CA Technologies product documentation, complete our short customer survey which is available on the CA Support website at http://ca.com/docs.

Contents

Chapter 1: Enhanced Features

The Release Notes for CA XCOM Data Transport for z/OS documents both new features and changes to existing features for r11.6.

This section contains the following topics:

XCOMUTIL History File Migration

The format of the VSAM history file for r11.6 has been modified. To continue to use an existing VSAM history file, perform a conversion of the file to generate a new history file for use with CA XCOM Data Transport r11.6.

More information:

- XCOMUTIL History Migration (see page 25)
- *CA XCOM Data Transport for z/OS User Guide*

zIIP Enablement

An important requirement today for any enterprise is the need to reduce costs that are associated with CPU utilization. CA XCOM Data Transport systems provide for the offloading of CPU intensive workloads to a zIIP processor to lower the cost of performing file transfers on the mainframe.

Data compression and de-compression functions, which by nature are CPU intensive, are offloaded to a zIIP processor when available and feasible. This function aids in reducing the cost of the transfer in terms of processor utilization.

More information:

- zIIP Enablement (see page 29)
- *CA XCOM Data Transport for z/OS User Guide*
- *CA XCOM Data Transport for z/OS Administration Guide*

Data Set Support

CA XCOM Data Transport has been enhanced to support additional types of data sets which have been introduced in recent releases of the z/OS operating system. Support is being provided for Large and Extended Format data sets. In addition support is provided for data sets residing in the extended address space of an Extended Address Volume.

More information:

- *CA XCOM Data Transport for z/OS User Guide.*

Large Format Data Sets

CA XCOM Data Transport supports large format data sets. These are physical sequential data sets which can have an allocation of more than 65,535 tracks on a single DASD volume. The maximum allocation for a large data set is 16,777,215 tracks.

Extended Format Data Sets

CA XCOM Data Transport supports extended format data sets. These are physical sequential data sets which are stored in a different internal format. This format supports data striping and compression. These data sets are created by specifying EXTREQ or EXTPREF for the DSNTYPE. EXTREQ specifies that the file must be allocated in extended format. EXTPREF specifies that the data set is preferred to be extended format, but will be allocated as a normal sequential data set if resources are not available. These data sets can only be allocated under SMS control as the striping and compression formats are defined in SMS classes.

Extended Address Volumes

CA XCOM Data Transport supports data sets in the Extended Address Space of an Extended Address Volume. The Extended Address Space is the space on a DASD volume beyond cylinder 65,519. Data sets residing in the extended address space are allocated in chunks of 21 cylinders and managed by cylinder. These data sets may also have extended attributes which are not available to data sets residing in track managed space below cylinder 65,520. Extended attributes are specified via the EATTR parameter. These data sets can only be allocated under SMS control as the extended attributes are defined in SMS classes. The data sets which may reside in the extended address space based on z/OS release are listed below.

z/OS 1.10

- VSAM
- zFS

z/OS 1.11

- VSAM
- zFS
- Extended format sequential

z/OS 1.12

- VSAM
- zFS
- All sequential data sets (Basic, Large, Extended)
- Partitioned data sets (PDS, PDSE)

SMF Records

Prior to release 11.6 of CA XCOM Data Transport, the SMF record written contained a subset of the information provided in the history record. For release 11.6, the SMF record written by CA XCOM Data Transport has been updated and expanded to include complete history record information. In addition, the size of the header for the SMF record has been increased by 6 bytes, currently unused, to allow the history data to start on a double word boundary.

More information:

- *CA XCOM Data Transport for z/OS Administration Guide.*

Trace Facility

The CA XCOM Data Transport Trace Facility has been enhanced for release 11.6 in order to improve usability of trace data. The enhancements are for the TRACE command and not the XTRACE command as the XCOMPLEX administrator function is deprecated as of release 11.6:

- Single output data set – Currently invoking the trace facility produces two distinct output data sets, TRACEDD and SYSPRINT. For this release all trace output will be consolidated into a single output data set, TRACEDD.
- Addition of 10 trace levels to control the amount of trace data produced. Valid trace levels are 0 through 9, with level 0 providing the minimal trace data and level 9 providing maximum trace data.

Format

```
F XCOM,TRACE[,{*|luname,[SNA]|ipname},[level]]
```

Operands

The following operands are used with the TRACE command:

*** (asterisk)**

Indicates that all active sessions involving this system are to be traced.

F XCOM,TRACE,* (with the asterisk operand) and

F XCOM,TRACE (without an operand) are equivalent.

***luname*,[SNA]**

Specifies the name (one to eight characters) of the LU to be traced. An LU name can be traced before a session exists. There is no requirement that the LU be defined to allow CA XCOM Data Transport to initiate a trace.

The optional keyword SNA should be used in cases where the LUNAME and IPNAME are the same at a site. This forces the LUNAME to be treated exclusively as SNA when creating the trace file.

ipname

Specifies the name or address of the IP destination to be traced.

level

Specifies the level of tracing desired. Valid levels are 0 through 9, with 0 providing the least and 9 providing the most trace data.

APF Authorization Required

Previous releases of CA XCOM Data Transport allowed jobs to be run without APF authorization, with certain features disabled. Beginning with this release, APF authorization is required to run CA XCOM Data Transport. APF authorization is being made mandatory as some of the new features for this release, and many features added in previous releases require APF authorization. The new features in this release which require APF authorization are zIIP Enablement and the new XCOMPLEX facility. Any programs that call XCOMJOB as an API linking with AC(1) is a must.

Changed Messages for APF Authorization

The XCOMM0049E (Not APF Authorized) message has been modified to remove the indication that some functions will not be available. This message will now indicate that the job is terminating.

0049E

XCOM IS NOT APF AUTHORIZED—JOB TERMINATING

Reason:

APF authorization is required to run CA XCOM Data Transport functions. The job is unable to continue and is terminated.

Action:

Authorize the CA XCOM Data Transport libraries.

For more information, see the *CA XCOM Data Transport for z/OS Message Reference Guide.*

Messages Removed for APF Authorization

The following messages which indicate that a function is not available due to lack of APF authorization are being removed. These messages are no longer relevant with the requirement of APF authorization for CA XCOM Data Transport.

0050E

xxxxxxxx NOT NOTIFIED—XCOM IS NOT APF AUTHORIZED

Reason:

The TSO user *xxxxxxxx* could not be sent a message that a file transfer completed because the CA XCOM Data Transport address space is not authorized (see the chapter "Installing Your Product" in the *CA XCOM Data Transport for z/OS Installation Guide*).

Note: This situation will *not* abort the file transfer.

Action:

CA XCOM Data Transport must be authorized if you require this function.

0051E

XCF FACILITY NOT STARTED—XCOM IS NOT APF AUTHORIZED

Reason:

All libraries must be APF authorized in order to use the IBM Coupling Facility. The XCOMPLEX requires the IBM Coupling Facility. XCOMPLEX servers will be started but without XCF facility.

Action:

Make sure all libraries are APF authorized.

0052E

APF AUTHORIZATION REQUIRED FOR SECURITY—XCOM TERMINATING

Reason:

Because the CA XCOM Data Transport address space is not authorized, the CA XCOM Data Transport security interface to the host computer was not available. For this reason, CA XCOM Data Transport ended.

Action:

If you require the use of its security interface, CA XCOM Data Transport must be authorized.

0055E

APF AUTHORIZATION IS REQUIRED TO TRANSFER USS FILES OR USE SECURITY=SAF

Reason:

Security calls to validate access to USS data sets must be made using SAF. Making SAF interface calls in these cases requires APF authorization.

Action:

Make the necessary changes to the processing environment so that the XCOMJOB program will run with APF authorization.

1004E

MUST RUN AUTHORIZED TO USE XCF SERVICES

Reason:

All libraries must be APF authorized in order to use the IBM Coupling Facility. The XCOMPLEX requires the IBM Coupling Facility.

Action:

Make sure all libraries are APF authorized.

PLEXQ Functional XCOMPLEX Replacement

The XCOMPLEX was introduced in CA XCOM Data Transport for z/OS release 3.1. It provided a means to generically schedule a transfer request to a group of servers. The PLEXQ facility introduced in r11.6 provides a second-generation implementation of this generic transfer scheduling facility.

Using the PLEXQ facility will provide better performance and conserve system resources as compared to the XCOMPLEX facility.

With the introduction of the PLEXQ facility, the XCOMPLEX environment (along with its ADMIN server) is deprecated. A PLEXQ environment may be introduced in servers which are also part of an XCOMPLEX. The two may coexist so that migration to the PELXQ may be accomplished more easily.

More information:

- PLEXQ Facility (see page 39)
- *CA XCOM Data Transport for z/OS User Guide*
- *CA XCOM Data Transport for z/OS Administration Guide*

CONFIG Member

CA XCOM Data Transport has a set of system parameters that govern its operation across all of its various interfaces. These system parameters have historically been kept in the CA XCOM Data Transport Default Options Table. Whenever a change is desired to the system parameters, it requires a link and reassemble of the Default Options Table. In CA XCOM Data Transport r11.6, these system parameters, Default Options, are now stored in a TYPE=CONFIG control library member.

The TYPE=CONFIG members replace the deprecated Default Options Table in this release. The migration from Default Options Table to TYPE=CONFIG member is simplified by an automatic conversion process the first time a Default Options Table is loaded by a server, admin server, or batch job. In addition, existing batch JCL with the DFLTAB parameter specified interprets the DFLTAB parameter as the CONFIG parameter, without failing the job.

For example, a TYPE=CONFIG member is automatically created in the first data set in the XCOMCNTL DD concatenation the first time a Default Options Table is used, for which no TYPE=CONFIG member exists. Subsequent server initialization using the same DFLTAB or CONFIG name uses the TYPE=CONFIG member instead of the assembled Default Options Table.

Default options parameters introduced in r11.6 are not accepted in assembly of the Default Options Table. The default options parameters can only be specified in the TYPE=CONFIG member.

More information:

- CONFIG Member (see page 51)
- *CA XCOM Data Transport for z/OS User Guide*

UNICODE and Multi-Byte Character Set Support for Data Transfer

CA XCOM Data Transport currently performs data transfers utilizing one of three data formats – ASCII, EBCDIC, or Binary. This enhancement allows for transmission of text files that are encoded using multi-byte character sets, including in-flight conversion of data between different character sets.

The IBM Unicode Services Facility is utilized for performing data conversion operations by CA XCOM Data Transport.

More information:

- Unicode and Multi-Byte Character Set Support for Data Transfer (see page 61)

Security Class Overrides

When performing partner and command resource security checks, CA XCOM Data Transport use to issue the security query against a specific class – FACILITY for partner resources and OPERCMDS for operator commands.

This enhancement allows an installation to specify the security class to issue the security calls against. Two new parameters have been added to the CA XCOM Data Transport configuration member. These parameters allow you to specify the security class to query for partner and command security.

The FACILITY parameter specifies the security class for partner resources and the OPERCMDS parameter specifies the security class for operator command resources. If these parameters are not specified in the configuration member, the security class will default to the CA XCOM Data Transport r11.5 values of FACILITY and OPERCMDS, respectively.

XCOMRRDS Format Modified

The format of the XCOMRRDS Request Queue File has been modified for Release 11.6. The maximum length of a request queue entry has been increased to 3030 bytes to allow for recording of additional transfer information. Create an XCOMRRDS Request Queue File using the Release 11.6 DEFRRDS sample described in the CA XCOM Data Transport User Guide.

The CA XCOM Data Transport Server (XCOMXFER) requires the XCOMRRDS Request Queue File. If the XCOMRRDS DD is not supplied or the file cannot be opened for processing by Release 11.6, the server terminates with RC=8.

XCOMRRDS Transfer Queue Startup Option

A new parameter, QSTART, has been added to provide greater control of the CA XCOM Data Transport server startup. There are three options which can be specified for the QSTART parameter:

- WARM
- COLD
- HOLD

New and Changed Messages

This section lists new/changed messages for this release that are not related to the new enhancements. Enhanced features have new and changed messages. For more information, see the chapter for each enhancement feature.

0076E

xxxxxxxx PARAMETER INVALID-USING DEFAULT VALUE

Reason:

An invalid parameter, xxxxxxxx, was specified on the EXEC statement PARM field for XCOMJOB or the CA XCOM Data Transport server.

Action:

For a description of the PARM keyword parameters, see the chapter The Batch Interface in the CA XCOM Data Transport for z/OS User Guide.

0137I

yyyyyyyyyyyyyyyyyy RECORDS SENT

Reason:

A file transfer completed successfully. The number of logical records transmitted was yyyyyyyyyyyyyyyyyy. Periods replace leading zeroes. In the history record and ISPF displays, leading zeroes are removed.

Action:

None

0147I

xxxxxxxxxxxxxxxxxx RECORDS RECEIVED

Reason:

A file transfer request with a remote computer has completed successfully. The total number of records received was xxxxxxxxxxxxxxxxxx. Periods replace leading zeroes. In the history record and ISPF displays, leading zeroes are removed.

Action:

None

0285E

XCOMRRDS DD STATEMENT MISSING

Reason:

CA XCOM Data Transport was about to open the XCOMRRDS Restart File but VSAM indicated that the DD statement was missing. The CA XCOM Data Transport Server terminates with RC=8.

Action:

Ensure that the XCOMRRDS DD statement is included in the CA XCOM Data Transport server JCL.

0286E

XCOMRRDS OPEN ERROR xx yy

Reason:

CA XCOM Data Transport was attempting to open the XCOMRRDS Restart File but VSAM returned an error code of xx yy. This return code is documented in the IBM documentation. The CA XCOM Data Transport Server terminates with RC=8.

Action:

None

0287E

XCOMRRDS NOT DEFINED AS RRDS

Reason:

CA XCOM Data Transport has successfully opened the XCOMRRDS Restart File but the file was not defined as an RRDS file. The CA XCOM Data Transport Server terminates with RC=8.

Action:

Ensure that the procedures for defining the RRDS file are followed correctly.

0288E

XCOMRRDS RECLEN INVALID—S/B xxxx BYTES

Reason:

CA XCOM Data Transport has successfully opened the XCOMRRDS Restart File but the record length that is defined for the file is not XXXX bytes. The CA XCOM Data Transport Server terminates with RC=8.

Action:

Ensure that the procedures for defining the XCOMRRDS Restart File are followed correctly.

0367I

xxxxxxxxxxxxxxxxxx BLOCKS yyyyyyyyyyyyyyyyyy RECS RECEIVED

Reason:

CA XCOM Data Transport has successfully completed a receive type file transfer. xxxxxxxxxxxxxxxxxx blocks and yyyyyyyyyyyyyyyyyy records were received. Periods replace leading zeroes. In the history record and ISPF displays, leading zeroes are removed.

Action:

None

0373I

xxxxxxxxxxxxxxxxxx BLOCKS yyyyyyyyyyyyyyyyyy RECS SENT

Reason:

CA XCOM Data Transport has successfully completed a send type file transfer. xxxxxxxxxxxxxxxxxx blocks and yyyyyyyyyyyyyyyyyy records were sent. Periods replace leading zeroes. In the history record and ISPF displays, leading zeroes are removed.

Action:

None

0479I

TRANSFER QUEUE BEING xxxx STARTED

Reason:

The CA XCOM Data Transport server is being started with the xxxx parameter for the QSTART option. The following explains the meaning of each possible QSTART option.

WARM: The XCOMRRDS transfer queue data set is being read sequentially, and all transfer requests that were pending when the server was last shutdown will be queued for execution with the same status values at the time the server was terminated.

COLD: All XCOMRRDS transfer request entries in the queue will be deleted, regardless of status. This has the same basic effect as reinitializing the XCOMRRDS data set.

HOLD: The XCOMRRDS transfer queue data set is being read sequentially, and all LOCALLY initiated transfer requests that were pending when the server was last shutdown will be placed in the server queue with a HOLD status.

Action:

None

0490E

RRDS DATASET IS IN USE BY ANOTHER TASK. RRDS=xx...xx

Reason:

The attempt to open the XCOMRRDS data set has failed with return code x'08' and fdbk2 code x'A8', which indicates that the data set is already open by another task. The value that is displayed for RRDS= is the name of the data set that was being opened. The CA XCOM Data Transport Server terminates with RC=8.

Action:

Ensure that the XCOMRRDS DD statement references the correct data set name, or identify the task that has it open and close it.

0639I

xxxxxxxx SNASVCMG SESSION DROPPING DUE TO DROPSESS OPTION

Reason:

Due to inactivity based on the timer setting of the DROPSESS configuration parameter, the SNASVCMG session for the specified LU has been dropped.

Action:

None.

0767I

ASSEMBLED DEFAULT OPTIONS TABLES ARE DEPRECATED AS OF r11.6

Reason:

Informational message indicating that a default options table was loaded on startup of the CA XCOM Data Transport server or admin, or as part of a batch job. A new configuration member replaces the use of the default option table and will be removed from the product in a future release.

Action:

None. The default options table is converted to a configuration member and resides in the first dataset that is defined as part of XCOMCNTL.

0768I

PLEASE REFER TO THE ADMINISTRATION GUIDE FOR INFORMAT ION REGARDING USAGE OF TYPE=CONFIG MEMBERS.

Reason:

Informational message indicating where to locate information about the new configuration members.

Action:

None.

1196I

THE XCOMXADM (ADMIN) SERVER HAS BEEN DEPRECATED AS OF r11.6

Reason:

Informational message indicating that the XCOMPLEX Admin Server is deprecated and replaced by PLEXQ which does not require an Admin server.

Action:

None.

1197I

PLEASE CONSULT THE ADMINISTRATION GUIDE FOR INFORMATION REGARDING USING PLEXQ FOR XCOMPLEX SERVICES

Reason:

Informational message indicating where to locate information about using PLEXQ for XCOMPLEX services.

Action:

None.

1426E

GETMAIN FAILED FOR PARAMETER PROCESSING

Reason:

Not enough storage is available to process SYSIN01 parameters.

Action:

Run the batch job in a larger region.

Chapter 2: XCOMUTIL History Migration

Modifications to the format of the VSAM history file cluster have been implemented for Release 11.6. This will require either defining a new VSAM history file cluster or migrating an existing one. The XCOMUTIL History File utility has been modified to perform a conversion of an existing release 11.5 VSAM history file cluster to the new 11.6 format. For more details refer to the CA XCOM Data Transport User Guide for the description of the XCOMUTIL utility. The format changes implemented for release 11.6 are as follows:

- The maximum length of the history record has been increased to 3030 bytes from 2020 in previous releases to allow for recording of additional transfer information.
- The alternate indexes have been made unique and will use the same date and time values as are in the primary key to enforce the uniqueness.
- A new unique alternate index based on the TCP/IP name has been added.

To use an existing VSAM history file from the 11.5 release in 11.6, this migration utility must be performed to create a new format history file. Unpredictable results will occur when using an 11.5 format VSAM history file with 11.6.

This section contains the following topics:

New and Changed Parameters

This section describes the new and modified parameter values allowed by the XCOMUTIL utility for performing VSAM history file migration in support of the 11.6 release.

PROCESS

A value of HISTORY_MIGRATION may be specified to invoke conversion of an existing VSAM history file from release 11.5 format to 11.6 format, and vice-versa.

RELEASE

This new parameter allows for specifying which release of CA XCOM Data Transport to convert the VSAM history file. Valid values are 11.5 and 11.6.

AGE

There are no changes to this parameter. This parameter can optionally be specified for history migration to remove old history records from the new format history file.

DATE

There are no changes to this parameter. This parameter can optionally be specified for history migration to remove old history records from the new format history file.

New and Changed Messages

This section describes the new and changed messages to support this enhancement.

Messages XCOMM8002E, XCOMM8005E, XCOMM8008E and XCOMM8010E have been modified to include new parameters added for the migration function. New messages for the migration function are XCOMM8018E, XCOMM8019E and XCOMM8020E. In addition, message XCOMM8017E was previously undocumented and is relevant for both the purge and migration functions.

8002E

INVALID PARAMETER CARD—NOT DATE/AGE, PROCESS OR RELEASE

Reason:

The only valid SYSIN statements for XCOMUTIL are DATE=, AGE=, PROCESS= and RELEASE=.

Action:

Remove the incorrect SYSIN statement and rerun XCOMUTIL.

8005E

PROCESS CARD DEFINED BUT OBJECT IS NOT HISTORY OR HISTORY_MIGRATION

Reason:

The SYSIN parameter PROCESS=HISTORY or PROCESS=HISTORY_MIGRATION was not specified correctly.

Action:

Specify PROCESS=HISTORY or PROCESS=HISTORY_MIGRATION and rerun XCOMUTIL.

8008E

NEITHER AGE NOR DATE CARDS ENTERED—ERROR

Reason:

No AGE or DATE SYSIN parameter was entered when PROCESS=HISTORY is specified.

Action:

Add an AGE or DATE parameter to the SYSIN parameters and rerun XCOMUTIL.

8010E

PROCESS CARD MISSING—XCOMUTIL TERMINATED

Reason:

The SYSIN parameter PROCESS was not specified.

Action:

The SYSIN parameter PROCESS=HISTORY or PROCESS=HISTORY_MIGRATION is required. Add the parameter and rerun XCOMUTIL.

8017E

DAYS ENTERED GREATER THAN THE NUMBER OF DAYS FOR THE SPECIFIED YEAR

Reason:

The day number specified in the Julian date is too large for the specified year. The value must not be greater than 365 for nonleap years, 366 for leap years.

Action:

Correct the Julian date specified for the DATE parameter and rerun XCOMUTIL.

8018E

INVALID RELEASE FOR MIGRATION SPECIFIED – MUST BE 11.5 OR 11.6

Reason:

When running a history migration, the RELEASE= parameter does not contain a valid release to migrate the history file cluster to.

Action:

Correct the RELEASE parameter to be 11.5 or 11.6 and rerun XCOMUTIL.

8019E

HISTORY FILE XXXXXXXX NOT AT PROPER RELEASE LEVEL FOR MIGRATION

Reason:

When running a history migration, the source history file sequential data set does not contain the proper size history records for the migration selected.

Action:

Correct the LRECL for the history file sequential data set created by the IDCAMS job to repro the history file cluster. The proper size of the history record is 2024 for an 11.5 history file or 3034 for an 11.6 history file and rerun XCOMUTIL.

8020E

MULTIPLE RELEASE PARAMETERS ENTERED

Reason:

Multiple RELEASE parameters were specified for the history file migration process.

Action:

Remove the extra RELEASE parameters and rerun XCOMUTIL.

Chapter 3: zIIP Enablement

zIIP is a special processor that is restricted to executing specific types of SRB mode work. zIIP is intended to free up general computing capacity and lower the overall cost of computing for CPU intensive workloads.

Release 11.6 of CA XCOM Data Transport allows for offloading workload onto zIIP processors to reduce CPU utilization costs that are associated with file transfers. Initially data compression and de-compression functions are offloaded to zIIP when available and feasible, as they are highly CPU intensive functions.

CA XCOM Data Transport attempts to offload data compression and de-compression functions to a zIIP when available. Given that data compression accounts for a significant portion of CPU utilization for a data transfer, moving this processing to a zIIP reduces utilization and costs.

Enabling CA XCOM Data Transport to utilize zIIP support requires that the installation has CA Common Services for z/OS installed at the r11 release or higher. An APAR is required to be installed for both the r11 and r12 releases of CA Common Services for z/OS which introduces the zIIP Enablement Service. For the r11 release, APAR RO27636 must be installed. For the r12 release, RO27110 must be installed. The common services library must be available to the XCOM Server started task or XCOMJOB TYPE=EXECUTE job.

By default, CA XCOM Data Transport enables zIIP support providing that the zIIP Enablement Service is available to load. A new PARM and configuration parameter can be specified to disable the zIIP support. The parameter is ZIIP=YES|NO, with YES being the default.

To insure that data compression functions are eligible to be moved to the zIIP without impacting performance. CA XCOM Data Transport requires that the size of the data block being compressed or de-compressed is a minimum of 4096 (4 K) bytes. This minimum size is to insure that the overhead of switching the workload to the zIIP does not affect performance of the transfer.

We recommend that the configuration parameter MAXPACK is set to a minimum of 4096. The higher the value for these parameters, the more efficient the use of zIIP is.

For any transfer, PACK=LENGTH is specified to insure that the 4-K minimum size for compression on the zIIP is realized. Without this parameter, the record size of data in a file would need to be 4 K to move the compression workload to the zIIP.

CA XCOM Data Transport provides some facilities for managing and monitoring zIIP usage. The only goal of utilizing zIIP is to reduce CPU utilization and therefore the costs that are associated with that utilization.

For each transfer, statistics about CPU and zIIP usage are written into the history record and SMF record. This data is retrieved from the zIIP Enablement Services component of CA Common Services. This data consists of six fields which can then be reported on through a TYPE=HISTORY job and supplied CA Easytrieve reports.

- Total CPU time used
- Total TCB mode CPU time
- Total SRB mode CPU time (This mode zIIP requires)
- Total zIIP CPU time
- Total zIIP eligible time that is spent on standard CPU
- Total time that is qualified to run on zIIP

A new modify command for the CA XCOM Data Transport Server is being provided to both report on status and allow zIIP support to be enabled/disabled.

The ZIIP,STATUS command provides information about the CPUs (standard and zIIP) as to how many are defined and online. The command also displays the status of zIIP support for the various features of CA XCOM Data Transport that can exploit use of zIIP. This command is the only data compression currently. Finally, statistics are displayed which show the total amount of processing time that was eligible to run on zIIP and that executed on zIIP.

The STATUS command is also issued on startup and termination of the CA XCOM Data Transport Server and XCOMJOB TYPE=EXECUTE start and termination.

The ZIIP,ENABLE and ZIIP,DISABLE commands allow an administrator to either enable or disable the zIIP support for future data transfers. Any transfers currently in progress continues to operate in the state that was active at the start of the transfer.

If an abend occurs while the transfer is running on a zIIP, CA XCOM Data Transport handles the abend. CA XCOM Data Transport also disables future use of zIIP, and restart the transfer from the last checkpoint. On the restart, zIIP is not used due to the disabling of the support.

The output of the SHOW command for a specific transfer displays the current zIIP CPU eligibility and utilization times.

This section contains the following topics:

New Parameters

This section describes the new PARM and Configuration parameter value allowed by the CA XCOM Data Transport Server and XCOMJOB TYPE=EXECUTE jobs for enabling /disabling zIIP support.

ZIIP

Specify YES or NO (Y or N are also acceptable) to indicate if zIIP support is to be enabled for the Server or XCOMJOB TYPE=EXECUTE Job. The default is YES.

New Commands

This section describes the new modify command to manage and monitor zIIP utilization by the Server.

ZIIP Command

Allows for management of zIIP services.

Format

```
F XCOM,ZIIP[,{ STATUS | ENABLE | DISABLE}]
```

Operand

The following operands may be used with the ZIIP command:

STATUS

Displays information pertaining to number and status of main CPU and zIIP processors.

Displays status of zIIP services for all CA XCOM Data Transport facilities that are zIIP aware.

Displays current statistics for the amount of processing time that was eligible to execute on zIIP and actually was executed on zIIP.

ENABLE

Will enable zIIP to be utilized for offloading processing from the main CPUs.

DISABLE

Will disable zIIP from being utilized to offload processing from the main CPUs.

Notes

Currently only the Compression and de-compression facilities of CA XCOM Data Transport support the offloading of processing to a zIIP.

Command Security Requirements

The use of the command requires that the command issuer's security authorization contain the following access level and the security class specified by the OPERCMDS parameter in the configuration member:

XCOM.*applsec*.ZIIP.*operand*

READ

XCOM.*applsec*.ZIIP.*STATUS*

UPDATE

XCOM.*applsec*.ZIIP.*ENABLE*

XCOM.*applsec*.ZIIP.*DISABLE*

The variables represent the following values:

applsec—the value of the APPLSEC parameter in the Default Options Table.

History Record Detail Screen

The History Record Detail screen provides information on the set of file transfers defined on the File Transfer Display Select screen. Additional statistical information will be presented to indicate CPU and zIIP processor utilization.

```
  CA XCOM SEND    FILE  REQ.#  001302 QUEUED MONDAY   AUG. 01, 2011    09:32:15

 COMMAND INPUT ===>

  Local System Identification
    Server: 1234XCOM                                  Port: 8040  Protocol: TCP

 History System ID: XC12  History System Name: XCM123  Invoking Job: USER01
 Sched. Start Time: MONDAY   AUG. 01 2011   09:32:19   Transfer ID: USER01234
          End Time: MONDAY   AUG. 01 2011   09:32:19   Encoding   : UTF-8
       Last Action: * NOT USED* Status: COMPLETED  Priority Sel: 016 Exec: 016
  Compress Mode: RLE       Trans. Time (Secs): 1    Compress Factr: 02.0
 Transfrd. Records: 1                  Bytes: 48
         Bytes/Sec: 1          Compress Bytes: 49
CPU: Time: 216,789                     zIIP:  Elig: 47,635
(ms)  TCB: 169,010                     (ms)   zIIP: 45,685
      SRB: 47,779                             CPU: 1,949

 Charset Input Error  : REPLACE          Replace Count: 0
 Charset Convert Error: REPLACE          Replace Count: 0

 Last Ms: XCOMM0137I 1 RECORDS SENT SUCCESSFULLY - FILE=XCOM.USERID01.MG

 ---------- S E N D I N G    S Y S T E M    I N F O R M A T I O N ------------
  System ID: *LOCAL*
    User ID: USER01                                    Notify ID: N/A
       Unit:                    Volume:                File Type: FLAT FILE
  File Name: XCOM.USER01.MGTCENT.TXT
  Charset  : CCSID#37/RE
  Rec Delim: EBCDIC:NA

 ---------- R E C E I V I N G    S Y S T E M    I N F O R M A T I O N --------
  System ID: USER01-LAPTOP
    User ID: user01                                      Notify ID: user01
       Unit:                    Volume:                  File Type: REPLACE
  File Name: c:\temp\mgtcent.txt

  Charset  : CCSID#850/ML
  Rec Delim: EBCDIC:CRLF:NL

F1=Help     F2=SPLIT     F3=End       F4=RETURN    F5=RFIND    F6=RCHANGE
F7=UP       F8=DOWN      F9=SWAP      F10=Unicode  F11=Hold    F12=Alloc
```

The following fields have been added to this screen:

Invoking Job

Specifies the job name that invoked the transfer.

CPU

Provide CPU utilization times for the file transfer. All times are reported in milliseconds (ms).

Time: Total CPU processing time for the transfer.

TCB: CPU processing time spent in TCB (task) mode.

SRB: CPU processing time spent in SRB (service) mode. This is the mode required to execute on a zIIP processor.

zIIP:

Provide zIIP utilization times for the file transfer. All times are reported in milliseconds (ms).

Elig: The amount of CPU time that was eligible to run on a zIIP.

zIIP: The amount of execution time on a zIIP.

CPU: The amount of eligible zIIP time that was executed on a normal CPU.

New Messages

This section describes the new messages added to support zIIP utilization.

0890I

zIIP PROCESSING HAS BEEN XXXXXXXX

Reason:

The XCOM Server or XCOMJOB TYPE=EXECUTE has started and is providing an indication of the state of zIIP processing. The state will be either ENABLED or DISABLED.

Action:

None. This message is displayed for informational purposes only.

0440I

zIIP cccccccc TIME=nnnnnnnnnnnnnnnnnnnnnnnnnnnnnnnn MICROSECONDS

Reason:

The user entered a SHOW,req# command. This message displays as part of the response to the SHOW command. This message displays the actual and eligible amount of time that the transfer spent on a zIIP processor.

Action:

None

0891I

XXX OF XXX XXXXXXXX CPUs ARE ONLINE IN LPAR

Reason:

Status of the CPUs for the LPAR is being provided, either as a result of the XCOM Server starting, XCOMJOB TYPE=EXECUTE or the ZIIP STATUS command. The message will display the number of standard and zIIP CPUs, along with the number of CPUs that are available.

Action:

None. This message is displayed for informational purposes only.

0892E

UNABLE TO RETRIEVE zIIP CPU INFO, RC=XXX

Reason:

The ZIIP STATUS command has failed to retrieve CPU status information. The API call to zIIP Enablement Services to retrieve CPU information has returned an error which is given as the return code. CA XCOM Data Transport will continue to operate, but will not attempt to move processing to a zIIP.

Action:

Keep the CA XCOM Data Transport log and SYSLOG and contact CA Technical Support.

0893E

UNABLE TO INITIALZE ZIIP PROCESSING, RETURN CODE=XX REASON CODE=XX

Reason:

An attempt was made to initialize zIIP Enablement Services to allow for CA XCOM Data Transport to offload processing to a zIIP. The return and reason codes are from the zIIP Enablement Service API. CA XCOM Data Transport will continue to operate, but will not attempt to move processing to a zIIP.

Action:

Keep the CA XCOM Data Transport log and SYSLOG and contact CA Technical Support.

0894E

zIIP ENABLEMENT IS NOT AVAILABLE IN COMMON SERVICES LIBRARY

Reason:

An attempt was made to load the zIIP Enablement Service module, which is part of the CA Common Services library has failed. CA XCOM Data Transport will continue to operate, but will not attempt to move processing to a zIIP

Action:

If you are running r11 of CA Common Services, please insure APAR RO27636 is applied. For r12 of CA Common Services, please insure APAR RO27110 is applied. Releases of CA Common Services below r11 do not support zIIP Enablement Services.

0895E

zIIP PROCESSING TERMINATED DUE TO ABEND - REASON CODE=XXXXXXXX

Reason:

A process that was running on a zIIP has abended. The reason code is returned from the zIIP Enablement Service. zIIP processing is disabled, and the transfer will be restarted from the point of failure and will not use zIIP,

Action:

Keep the CA XCOM Data Transport log and SYSLOG and contact CA Technical Support.

0896I

zIIP PROCESSING FOR XXXXXXXXXXXXXXXX IS XXXXXXXX

Reason:

Issued as a result of the ZIIP command, status of each feature which can utilize zIIP processing is displayed.

Action:

None. This message is displayed for informational purposes only.

0897I

TOTAL XXXXXXXX zIIP TIME = XXXXXXXXX MICROSECONDS

Reason:

Issued as a result of a ZIIP STATUS command, termination of the XCOM Server or termination of XCOMJOB TYPE=EXECUTE, provides statistical information as to the amount of processing time that was both eligible to be run on a zIIP and actually took place on a zIIP.

Action:

None. This message is displayed for informational purposes only.

Chapter 4: PLEXQ Facility

CA XCOM Data Transport has implemented a means of generically scheduling to a group of servers. This was introduced as the XCOMPLEX. The PLEXQ facility is the redesign of that facility to provide enhanced performance and system resource utilization.

Release 11.6 of CA XCOM Data Transport will provide this enhanced PLEXQ capability.

IBM SYSPLEX Signaling Services are the means of communication in the new PLEXQ architecture. Batch XCOMJOB utility jobs connect to a PLEXQ group, to which CA XCOM Data Transport servers are already joined. The XCOMJOB utility communicates with the member servers in the PLEXQ to manage transfer workload and activity.

The PLEXQ brings with it a new protocol which may be substituted for SNA and TCP/IP when performing meta-transfers between XCOMJOB and XCOM servers. A PLEXQ group can be used to communicate with one or many servers, thus it can be used as an alternate protocol for performing transfer scheduling and inquiry for even an individual server. The advantage of using the PLEXQ is that it uses less storage and requires fewer system resources than when an SNA or TCP/IP connection is used.

PLEXQ connectivity can be used for each of the following XCOMJOB request types:

- SCHEDULE
- INQUIRE
- HISTORY
- OPER
- RECOVER

Tracing has been enhanced in CA XCOM Data Transport to show the exchange of data between members of a PLEXQ group. Entries in the TRACEDD output dataset are created to show the state of communications as well dumps of buffer content to facilitate problem determination related to PLEXQ activities.

This section contains the following topics:

New Parameters

This section describes the new PARM and Configuration parameter value allowed by the CA XCOM Data Transport Server and XCOMJOB meta-transfer jobs for enabling PLEXQ support.

PLEXQ

Specify the name of the PLEXQ group to which the CA XCOM Data Transport server (XCOMXFER) is to connect. This parameter only applies to the server.

STCPLEXQ

Specify the name of the PLEXQ group to which the XCOMJOB batch utility is to connect. This parameter only applies to XCOMJOB executions using TYPEs other than EXECUTE. There must be a server already connected to the PLEXQ group in order for the request to complete successfully.

New Commands

This section describes the new modify commands to manage and monitor activities within PLEXQ member servers.

STAT Command

Allows for the display of task levels in servers which are members of the same PLEXQ as the server from which the command is issued.

Note: This STAT command is analogous to the STAT command which previously could only be issued from an XCOMPLEX ADMIN server. The PLEXQ STAT command can be entered from any peer server in the PLEXQ group.

Format

```
F XCOM,STAT
```

Command Security Requirements

The use of the STAT command requires that the command issuer's security authorization contain READ access to the following resource specification for CLASS(OPERCMDS):

XCOM.*applsec*.STAT

XSHOW Command

Allows for the display of locally initiated transfers in all members of the same PLEXQ as the server from which the command is issued.

Note: This XSHOW command is analogous to the XSHOW command which previously could only be issued from an XCOMPLEX ADMIN server. The PLEXQ XSHOW command can be entered from any peer server in the PLEXQ group.

Format

```
F XCOM,XSHOW
```

Command Security Requirements

The use of the XSHOW command requires that the command issuer's security authorization contain READ access to the following resource specification for CLASS(OPERCMDS):

XCOM.*applsec*.XSHOW

XRSHOW Command

Allows for the display of remotely initiated transfers in all members of the same PLEXQ as the server from which the command is issued.

Note: This XRSHOW command is analogous to the XRSHOW command which previously could only be issued from an XCOMPLEX ADMIN server. The PLEXQ XRSHOW command can be entered from any peer server in the PLEXQ group.

Format

```
F XCOM,XRSHOW
```

Command Security Requirements

The use of the XRSHOW command requires that the command issuer's security authorization contain READ access to the following resource specification for CLASS(OPERCMDS):

XCOM.*applsec*.XRSHOW

New Messages

1100E

PLEXQ IS DAMAGED. PLEXQ PROCESSING DISABLED.

Reason:

An IBM Signaling Services macro has terminated with a return code indicating that the PLEXQ GROUP has been damaged.

Action:

Terminate all CA XCOM servers and batch utilities that are connected to the PLEXQ group and attempt to restart them. In the event that this condition should occur, the underlying cause of the failure should be investigated. This is a system environmental error and a solution should be pursued through IBM support.

1101I

CONNECTED TO PLEXQ GROUP yyyyyyyy AS zzzzzzzz

Reason:

This message indicates that this CA XCOM program has connected to the PLEXQ GROUP identified in the message. Variable substitution in the message text is as follows:

yyyyyyyy = the name of the PLEXQ group to which this program connected.

zzzzzzzz = the name by which this program is know in the PLEXQ group.

Action:

None

1102I

DISCONNECTED FROM PLEXQ GROUP yyyyyyyy AS zzzzzzzz

Reason:

This message indicates that this CA XCOM program has disconnected from the PLEXQ GROUP identified in the message. Variable substitution in the message text is as follows:

yyyyyyyy = the name of the PLEXQ group to which this program connected.

zzzzzzzz = the name by which this program is know in the PLEXQ group.

Action:

None

1104I

wwwwwwww / xxxxxxxx CONNECTED TO PLEXQ GROUP yyyyyyyy AS zzzzzzzz

Reason:

This message indicates that another CA XCOM program has connected to the PLEXQ GROUP identified in the message, and to which the region which issued the message was already connected. Variable substitution in the message text is as follows:

wwwwwwww = the name of the system where the connector is running.

xxxxxxxx = the name of the job or started task which connected.

yyyyyyyy = the name of the PLEXQ group to which the program connected.

zzzzzzzz = the name by which the program is know in the PLEXQ group.

Action:

None

1105I

xxxxxxxx / yyyyyyyy DISCONNECTED FROM PLEXQ GROUP zzzzzzzz

Reason:

This message indicates that another CA XCOM program has disconnected from the PLEXQ GROUP identified in the message. The region issuing this message is connected to the named PLEXQ GROUP. Variable substitution in the message text is as follows:

xxxxxxxx = the name of the system where the disconnector is running.

yyyyyyyy = the name of the job or started task which disconnected.

zzzzzzzz = the name of the PLEXQ group to which the program connected.

Action:

None

1106E

THIS XCOM XXXXXX IS NOT RELEASE COMPATIBLE WITH PLEXQ GROUP YYYYYYYY

Reason:

An XCOM server attempted to connect to a PLEXQ where the other XCOM servers in the PLEX are at a different release of CA XCOM Data Transport. The PLEXQ only allows connections by servers running the same release of CA XCOM Data Transport. XXXXXX is the type of CA XCOM Data Transport job, SERVER, JOB or, ADMIN. YYYYYYYY specifies the name of the PLEXQ group that the connection is attempted to.

Action:

None required. The XCOM Server continues to run as a stand-alone server.

1113E

xxxxxxxx REQUEST FAILED RC=yyyyyyyy REASON=zzzzzzzz

Reason:

This message indicates that the macro specified in the message has completed with a non-zero return code, indicating that a failure has occurred. Variable substitution in the message text is as follows:

xxxxxxxx = the name of the macro which failed

yyyyyyyy = the return code from the failing macro call

zzzzzzzz = the reason sub-code from the failing macro call

Action:

Locate the RETURN and REASON codes for the appropriate macro in the IBM z/OS documentation and follow the instructions for the specific error code listed there.

1114E

PLEXQ TIMEOUT, ABORTING #RECEIVE OF RESPONSE

Reason:

This message indicates that an expected response to a prior request was not received by the PLEXQ protocol management module within the timeout limit.

Action:

Ensure that the partner region receiving the request is still operational. Increase the TIMEOUT parameter for the local region if the partner region is still functional. If all regions are still operational, gather all output from the requesting CA XCOM server or XCOMJOB batch utility and contact CA XCOM support.

1115E

xxxxxxxx – UNKNOWN MESSAGE TYPE IN CONTROL DATA

Reason:

A PLEXQ message which was received contains an invalid message type identifier. This message indicates an internal PLEXQ error.

Action:

Gather all output from the failing CA XCOM server or XCOMJOB batch utility and contact CA XCOM support.

1116E

xxxxxxxx – MISSING CSA ADDRESS (ZERO)

Reason:

A PLEXQ request has been received which is missing a critical control block address. This message indicates an internal PLEXQ error.

Action:

Gather all output from the failing CA XCOM server or XCOMJOB batch utility and contact CA XCOM support.

1117E

xxxxxxxx – CSA ADDRESS DOES NOT VERIFY

Reason:

A PLEXQ request has been received which has an invalid address for a critical control block. This message indicates an internal PLEXQ error.

Action:

Gather all output from the failing CA XCOM server or XCOMJOB batch utility and contact CA XCOM support.

1118E

xxxxxxxx – MISSING NSA ADDRESS (ZERO)

Reason:

A PLEXQ request has been received which is missing a critical control block address. This message indicates an internal PLEXQ error.

Action:

Gather all output from the failing CA XCOM server or XCOMJOB batch utility and contact CA XCOM support.

1119E

xxxxxxxx – NSA ADDRESS DOES NOT VERIFY

Reason:

A PLEXQ request has been received which has an invalid address for a critical control block. This message indicates an internal PLEXQ error.

Action:

Gather all output from the failing CA XCOM server or XCOMJOB batch utility and contact CA XCOM support.

1120E

xxxxxxxx – MISSING PLX ADDRESS (ZERO)

Reason:

A PLEXQ request has been received which is missing a critical control block address. This message indicates an internal PLEXQ error.

Action:

Gather all output from the failing CA XCOM server or XCOMJOB batch utility and contact CA XCOM support.

1121E

xxxxxxxx – PLX ADDRESS DOES NOT VERIFY

Reason:

A PLEXQ request has been received which has an invalid address for a critical control block. This message indicates an internal PLEXQ error.

Action:

Gather all output from the failing CA XCOM server or XCOMJOB batch utility and contact CA XCOM support.

1122E

xxxxxxxx - INSUFFICIENT STORAGE TO ATTACH THE REQUEST ED TRANSACTION PROGRAM

Reason:

There is not enough available virtual storage in the CA XCOM region to invoke the transaction program needed to service the request.

Action:

Do one or more of the following:

- Restart the CA XCOM region with more virtual storage, then resubmit the request.
- Reduce the number of concurrent tasks (MAXTASK, MAXREM and/or MAXLOC) so that more virtual storage is available for initiating internal requests within the CA XCOM region, then resubmit the request once some of the currently running tasks have completed.
- Resubmit the request at a later time when the CA XCOM region is less busy.

1123E

xxxxxxxx – INVALID ECB ADDRESS IN TRANSACTION PROGRAM REQUEST/RESPONSE CONTROL DATA

Reason:

The address of an ECB needed to communicate with another task within the CA XCOM region is not valid. This indicates an internal PLEXQ error.

Action:

Gather all output from the failing CA XCOM server or XCOMJOB batch utility and contact CA XCOM support.

1124E

COMMAND REJECTED - PLEXQ NOT ACTIVE

Reason:

A STAT, USTAT, XRSHOW, or XSHOW command was entered but the PLEXQ group is not active.

Action:

None.

1168I

PLEXQ LISTENER TASK {STARTING | STOPPING}

Reason:

The task within the CA XCOM region which processes PLEXQ messages is being started or is stopping.

Action:

None

1182E

INVALID PLEXQ COMMAND – INTERNAL ERROR

Reason:

The command byte in the internal PLEXQ request buffer is not valid.

Action:

Gather all output from the failing CA XCOM server or XCOMJOB batch utility and contact CA XCOM support.

1183E

xxxxxxxx REQUEST ENDED DUE TO PLEXQ ERROR

Reason:

An error occurred which caused the request identified by xxxxxxxx to not be completed successfully.

Action:

Review the XCOMLOG and other SYSOUT messages from the CA XCOM invocation looking for other error messages relating to the failed request. If insufficient information was displayed to identify the problem, gather all output from the failing CA XCOM server or XCOMJOB batch utility and contact CA XCOM support.

1184E

GETMAIN FAILED FOR PLEXQ BUFFER

Reason:

An attempt to acquire storage to use as a buffer for a PLEXQ operation has failed.

Action:

The most common cause of this message is a shortage of virtual storage available within the CA XCOM address space. If possible, increase the REGION JCL parameter for the execution of the program. If there is sufficient storage, check the CA XCOM SYSOUT for other messages relating to errors that may have occurred during the execution of the CA XCOM program.

1193E

ERROR JOINING PLEXQ GROUP xxxxxxxx

Reason:

The macro call issued to connect to the SYSPLEX Signaling Services Group (named by xxxxxxxx) has received a return code indicating that the connection attempt was unsuccessful.

Action:

Examine the XCOMLOG and SYSOUT from the CA XCOM region to locate any additional messages relating to the PLEXQ connection. Take the action(s) indicated for any such messages.

1194E

STACK EXPANSION (LSEXPAND) FAILED. RC=xxxxxxxx

Reason:

The LSEXPAND macro is used to expand the available stack area for program use within an address space. This macro call has failed with the return code indicated by xxxxxxxx.

Action:

Review the documentation for the LSEXPAND system service and follow the instructions for the return code that was received. This is likely a system environmental error.

1195E

NO SERVERS ARE CURRENTLY JOINED TO PLEXQ GROUP xxxxxxxx

Reason:

The XCOMJOB batch utility was invoked specifying a STCPLEXQ parameter of xxxxxxxx, but no servers were found connected to SYSPLEX Signaling Services using that group name.

Action:

Verify that the spelling of the group name on the STCPLEXQ EXEC PARM for the XCOMJOB utility is correct. If the group name is spelled correctly on the STCPLEXQ EXEC PARM, verify that a CA XCOM server (XCOMXFER) has been started, connected and is still active using the specified PLEXQ group name.

Allowing TYPE=OPER (Operator) Requests from ISPF to the PLEXQ

To allow users to perform TYPE=OPER transfers of PDSE program libraries, add XCOMPLEX to the AUTHPGM and the AUTHTSF tables of IKJTSO00 module in SYS1.PARMLIB. The CA XCOM Data Transport libraries used in your CLIST for the CA XCOM Data Transport ISPF interface must all be APF authorized also.

You can also refresh the TSO library using the TSO UPDATE PARMLIB(00) member. For more information, see the *IBM TSO/E Customization manual*.

Chapter 5: CONFIG Member

CA XCOM Data Transport has a set of system parameters that govern its operation across all of its various interfaces. The CA XCOM Data Transport system parameters take effect as soon as CA XCOM Data Transport is started and, unless overridden, they remain in effect as long as CA XCOM Data Transport is active. These system parameters have historically been kept in the CA XCOM Data Transport Default Options Table. These parameters are now stored in a TYPE=CONFIG control library PDS member.

Important! A TYPE=CONFIG member is automatically created in the first data set in the XCOMCNTL DD concatenation the first time a Default Options Table is used, for which no TYPE=CONFIG member exists. Subsequent server initialization using the same DFLTAB or CONFIG name uses the TYPE=CONFIG member instead of the assembled Default Options Table.

When migrating from r11.5 or below, it is mandatory that the Default Options Table is reassembled before starting a CA XCOM Data Transport address space for the first time. Reassembling the table ensures that default values for any new Default Option Table parameters are included in the creation of the TYPE=CONFIG member. Failure to reassemble the Default Option Table results in the following error message:

XCOMM002IE DEFAULT OPTIONS TABLE SHOULD BE REASSEMBLED.

GEN-LEVEL/VERSION (xxxx-xxx) DOES NOT MATCH XCOMXFER (yyyy-yyy).

The CA XCOM Data Transport address space is then terminated.

Note: For descriptions of CA XCOM Data Transport system parameters, see the Configuration Parameters chapter in the *CA XCOM Data Transport for z/OS Administration Guide*.

This section contains the following topics:

New Parameter

This section describes the new PARM parameter.

CONFIG

Indicates the name of the TYPE=CONFIG control library member that is read and used by the CA XCOM Data Transport server.

XCOMDFLT

Specifies the TYPE=CONFIG member XCOMDFLT.

xxxxxxxx

Specifies the name of the TYPE=CONFIG member other than XCOMDFLT. This name can be up to eight alphanumeric characters long.

Notes:

- If both the DFLTAB and CONFIG parameters are specified, the CONFIG parameter takes precedence.
- Using the CONFIG parameter assumes that the member named is an existing, valid TYPE=CONFIG member.

Default: XCOMDFLT

New Messages

1400I

Parameters from TYPE=CONFIG member XXXXXXXX

Reason:

This message identifies the source of the parameters that follow as being the TYPE=CONFIG member read during the XCOM server startup. XXXXXXXX is replaced by the name of the PDS member that was specified on the CONFIG= EXEC parameter.

Action:

None

1401I

Parameters from EXEC PARM= override

Reason:

This message identifies the source of the parameters that follow as being the EXEC statement PARM= field.

Action:

None

1405E

VALUE CONTAINS NON-NUMERIC DATA

Reason:

A parameter value was parsed which was expected to be numeric, but non-numeric data was found.

Action:

Correct the parameter value by removing any non-numeric characters and resubmit the job.

1406E

VALUE LOWER THAN MINIMUM ALLOWED

Reason:

The numeric value parsed from the parameter is less than the minimum allowed for this numeric parameter.

Action:

Correct the parameter value by changing it to conform to the prescribed limits.

1407E

VALUE HIGHER THAN MAXIMUM ALLOWED

Reason:

The numeric value parsed from the parameter is greater than the maximum allowed for this numeric parameter.

Action:

Correct the parameter value by changing it to conform to the prescribed limits.

1408E

VALUE LONGER THAN MAXIMUM ALLOWED

Reason:

The length of the value parsed from the parameter is greater than the maximum number of characters allowed for this parameter.

Action:

Correct the parameter value by changing it to conform to the prescribed limits.

1409E

VALUE SHORTER THAN MINIMUM ALLOWED

Reason:

The length of the value parsed from the parameter is less than the minimum number of characters allowed for this parameter.

Action:

Correct the parameter value by changing it to conform to the prescribed limits.

1410E

VALUE CONTAINS INVALID CHARACTERS

Reason:

The value parsed from the parameter contains characters that are not allowed for this particular parameter.

Action:

Correct the parameter value by changing it to conform to the prescribed format.

1411E

VALUE NOT RECOGNIZED FOR THIS PARAMETER

Reason:

The value parsed from the parameter does not match the keywords allowed to be specified for it.

Action:

Correct the parameter value by changing it to be a keyword from the list of possible options for this parameter.

1412E

PARAMETER NOT RECOGNIZED

Reason:

The keyword parameter specified in the record is not from the list of supported parameters which can be specified in a TYPE=CONFIG member.

Action:

Check the spelling of the keyword parameter and correct if necessary, or ensure that the parameter is in fact valid for CA XCOM.

1413E

INITIALIZATION TERMINATED DUE TO PARAMETER ERRORS

Reason:

Severe errors were detected in the parameter records in the TYPE=CONFIG member such that initialization of the CA XCOM program being invoked cannot continue.

Action:

Correct the parameter errors noted prior to this message and resubmit the job.

1414E

INITIALIZATION TERMINATED DUE TO PARAMETER WARNINGS AND A SETTING OF PARSE_MODE=STRICT

Reason:

Moderate errors were detected in the parameter records in the TYPE=CONFIG member, however the PARSE_MODE option is set to "STRICT". As such, initialization of the CA XCOM program being invoked cannot continue.

Action:

Correct the parameter errors noted prior to this message and resubmit the job.

1415E

DATA SET NAME OR PREFIX IS NOT VALID

Reason:

The data set name or prefix specified in the parameter being processed has failed the validation checks implemented by the CA XCOM parameter parsing program. These checks enforce the data set naming conventions established by IBM for z/OS data set names.

Action:

Correct the data set name or prefix specification on the parameter statement and resubmit the job.

1416E

CONTINUATION STATEMENTS MUST BEGIN IN COLUMNS 2 THROUGH 16

Reason:

The prior statement indicated it was being continued to the statement immediately preceding this message. However, the data contained in this parameter statement does not begin in the columns (2 – 16) which are required for second and subsequent continuation statements.

Action:

Check the starting column of this parameter statement and ensure that 1) continuation is desired, and 2) that the data in this statement begins in the correct column range required for continuation. Make any necessary corrections to the parameter statement and resubmit the job.

1417E

EXPECTED CONTINUATION STATEMENT MISSING

Reason:

The prior statement indicated it was being continued to the statement immediately preceding this message. However, the data contained in this parameter statement begins in column 1, which is not valid for a continuation statement.

Action:

Check the starting column of this parameter statement and ensure that 1) continuation is desired, and 2) that the data in this statement begins in the correct column range required for continuation. Make any necessary corrections to the parameter statement and resubmit the job.

1418E

END OF FILE REACHED, BUT CONTINUATION EXPECTED

Reason:

The prior statement indicated it was being continued to the next parameter statement. However, an end-of-file condition was raised before the continuation statement was read.

Action:

Ensure that the previous statement should have been flagged for continuation. If so, add a parameter statement to provide the continuation data that is expected. If not, remove the "+" from the prior parameter statement. Resubmit the job.

1419E

INVALID FIRST CHARACTER - MUST BE @, #, $ OR A - Z

Reason:

Some configuration parameters require that the first character be Alphabetic (A-Z) or the characters "@", "#" or "$". The parameter being defined has that requirement, but the value being provided does not conform to it.

Action:

Ensure that the first character of the value (or of each data set name node in the case of a data set name or prefix) conforms to the restriction outlined above.

1421E

MEMBER NAME IS REQUIRED FOR THE SAVE COMMAND

Reason:

The CA XCOM Data Transport SAVE console command was entered, but no member name was provided as a parameter.

Action:

Specify the name of the CONFIG member which is to be saved or replaced in the first XCOMCNTL data set.

1422E

xxxxxxxx IS AN INVALID OPTION FOR THE SAVE COMMAND

Reason:

The CA XCOM Data Transport SAVE console command was entered along with a member name, but the parameter identified in the message is not recognized as being valid for the command.

Action:

Either enter the console command again without the parameter noted in the message, or correct it to be a valid option for the SAVE command.

1423I

xxxxxxxx CONFIG MEMBER BEING CREATED [FROM DEFAULT TABLE xxxxxxxx GENERATED xxxxxxxx AT xx.xx]

Reason:

A CONFIG member with the name specified in the message is being created in the XCOMCNTL data set. The second, optional portion of this message is written if the member is being automatically created from an existing CA XCOM Data Transport Default Options Table, and includes the name and timestamp from when it was assembled.

Action:

None

1424E

xxxxxxxx MEMBER ALREADY EXISTS. USE REPLACE OPTION TO OVERWRITE IT.

Reason:

A SAVE command was entered for the module named in the message, but it already exists in the XCOMCNTL dataset. An existing CONFIG member can only be overwritten by adding the REPLACE optional keyword to the command syntax.

Action:

Verify that the member referenced in the message is the correct one, and either specify a different member name, or specify the REPLACE option to cause it to be overwritten.

1425I

xxxxxxxx CONFIG MEMBER BEING REPLACED

Reason:

The CONFIG member named in the message is being overwritten with the system parameters currently in effect.

Action:

None

1428I

PARAMETER XXXXXXXXXXXXXXXX HAS BEEN DEPRECATED, VALUE IGNORED

Reason:

The identified parameter was specified but the value will be ignored because the parameter has been deprecated.

Action:

Remove the identified deprecated parameter; otherwise the message can be ignored.

Chapter 6: Unicode and Multi-Byte Character Set Support for Data Transfer

Before the advent of Unicode, a significant number of character sets were devised to permit the representation of symbols used in the Chinese, Japanese, Korean, and Taiwanese (CJK) languages. Today, Unicode is favored and there is an ongoing transition from these legacy character sets to Unicode encodings, most notably UTF-8 and UTF-16.

Many CJK legacy multibyte character sets are ASCII based, as is the case for the most commonly used Unicode encodings (i.e.UTF-8, UTF-16).

In the IBM mainframe (predominantly EBCDIC) world however composite character sets are commonly employed, involving a Shift-in/Shift-out encoding method. This encoding mechanism enables a single-byte ASCII or EBCDIC character-set to be used for the representation of Latin characters, in tandem with a multibyte character set for the representation of non-Latin characters. Shift-in and shift-out control characters are then inserted in the data stream to signal a switch between the two embedded character sets. The CCSID 937 character set combines an EBCDIC single byte character-set with a Traditional Chinese multibyte character set. While the CCSID 938 character set combines an ASCII single byte character-set with the same Traditional Chinese multibyte character set.

CA XCOM Data Transport currently performs data transfers utilizing one of three data formats – ASCII, EBCDIC, or Binary.

This enhancement allows for transmission of text files that are encoded using multi-byte character sets, including in-flight conversion of data between different character sets. Two additional data formats can be specified for the CODE parameter to allow for transmission of these files. In addition, new parameters have been added to the CA XCOM Data Transport configuration member, destination member and SYSIN01 parameters. These parameters allow you to specify the local and remote character sets to be used for file data conversion and actions for dealing with unconvertible characters.

CA XCOM Data Transport is utilizing IBM Unicode Services to perform data conversion functions. For information on IBM Unicode Services, please refer to the IBM manual Unicode Services User's Guide and Reference, SA22-7649. It can be found on the IBM website.

The CODE SYSIN01 parameter allows for two new data formats – UTF8 and UTF16. When one of these formats is specified for a transfer, data is converted to that format for transmission to the remote partner.

The LOCAL_CHARSET and REMOTE_CHARSET SYSIN01 parameters specify the character-set of the local and remote files for the transfer. These parameters are used in conjunction with CODE=UTF8 or CODE=UTF16 to perform the conversion of data. If not specified for the transfer, they default to the value specified for the DEFAULT_CHARSET parameter in the destination or configuration member.

In order to handle conversion issues between character sets, additional parameters MBCS_INPUTERROR and MBCS_CONVERROR specify what action is taken in the event of a character being encountered cannot be converted. The sending partner uses MBCS_INPUTERROR and specifies to either replace the character with the replacement character defined in the IBM conversion table or fail the transfer. The receiving partner uses MBCS_CONVERROR and specifies to either replace the character with the replacement character defined in the IBM conversion table or fail the transfer. If not specified the value of DEFAULT_INPUTERROR and DEFAULT_CONVERROR parameters in the destination or configuration member will be used.

For USS files, SYSIN01 parameters LOCAL_DELIM and REMOTE_DELIM specify the encoding scheme that the corresponding character-set uses and a list of delimiters which exists within the data as record separators.

For receive requests where the target character set is different from the source character set, the LRECL needs to have a value specified which allows for the difference in the number of bytes per character. If the LRECL is not large enough to support the target character set, an XCOMM0144E SENDING RECLEN > MAX TARGET LENGTH error is issued.

For SEND JOB requests to a z/OS partner, the remote CCSID (REMOTE_CHARSET) is required to be EBCDIC based as JES is unable to process non-EBCDIC characters.

The use of truncation (TRUNCATE=YES) is not supported for Unicode transfers. This is due to the possibility of data loss or corruption should truncation occur in the middle of a multi-byte character in the file.

This section contains the following topics:

New Parameters

This section describes the new PARM and Configuration parameter values the CA XCOM Data Transport Server allows and XCOMJOB TYPE=EXECUTE jobs for Unicode support.

DEFAULT_CHARSET

Specify the CCSID to use for files when a character set is not specified in the transfer parameters for the local or remote file. Optionally specify the search order of conversion techniques to use for data conversion by the IBM Unicode Services. The default CCSID is 37 for US EBCDIC.

DEFAULT_DELIM

Specify the encoding scheme of the data and record delimiters to use for character conversion to and from Unicode format. The encoding scheme is relevant for USS files being transferred. The default value is EBCDIC:NL, which indicates EBCDIC data encoding and new line character for record separators.

DEFAULT_CONVERROR

Specify the action for IBM Unicode Services to take when a character encountered for conversion by the receiving partner is not included within the output character sets character repertoire. The only supported options are REPLACE and FAIL, with FAIL being the default. If FAIL is specified, the transfer fails. The transfer also fails when an unconvertible character is detected.

DEFAULT_INPUTERROR

Specifies the action for IBM Unicode Services to take when a character encountered for conversion by the sending partner is not consistent with the specified input character set. The only supported options are REPLACE and FAIL, with FAIL being the default. If FAIL is specified, the transfer fails. The transfer also fails when an unconvertible character is detected.

History Record Detail Screen

The History Record Detail screen provides information on the set of file transfers that are defined on the File Transfer Display Select screen. Additional information for Unicode transfers is presented.

```
  CA XCOM SEND    FILE  REQ.#  001302 QUEUED MONDAY   AUG. 01, 2011    09:32:15

 COMMAND INPUT ===>
  Local System Identification
    Server: 1234XCOM                                    Port: 8040  Protocol: TCP

 History System ID: XC12  History System Name: XCM123  Invoking Job: USER01
 Sched. Start Time: MONDAY   AUG. 01 2011   09:32:19   Transfer ID: USER01234
          End Time: MONDAY   AUG. 01 2011   09:32:19   Encoding   : UTF-8
       Last Action: * NOT USED* Status: COMPLETED  Priority Sel: 016 Exec: 016
  Compress Mode: RLE       Trans. Time (Secs): 1    Compress Factr: 02.0
 Transfrd. Records: 1                  Bytes: 48
         Bytes/Sec: 1          Compress Bytes: 49
CPU: Time: 216,789                     zIIP:  Elig: 47,635
(ms)  TCB: 169,010                     (ms)   zIIP: 45,685
      SRB: 47,779                             CPU: 1,949

 Charset Input Error  : REPLACE          Replace Count: 0
 Charset Convert Error: REPLACE          Replace Count: 0

 Last Ms: XCOMM0137I 1 RECORDS SENT SUCCESSFULLY - FILE=XCOM.USERID01.MG

 ---------- S E N D I N G    S Y S T E M    I N F O R M A T I O N ------------
  System ID: *LOCAL*
    User ID: USER01                                      Notify ID: N/A
       Unit:                    Volume:                  File Type: FLAT FILE
  File Name: XCOM.USER01.MGTCENT.TXT
  Charset  : CCSID#37/RE
  Rec Delim: EBCDIC:NA

 ---------- R E C E I V I N G    S Y S T E M    I N F O R M A T I O N --------
  System ID: USER01-LAPTOP
    User ID: user01                                         Notify ID: user01
       Unit:                    Volume:                     File Type: REPLACE
  File Name: c:\temp\mgtcent.txt

  Charset  : CCSID#850/ML
  Rec Delim: EBCDIC:CRLF:NL

F1=Help     F2=SPLIT     F3=End       F4=RETURN    F5=RFIND    F6=RCHANGE
F7=UP       F8=DOWN      F9=SWAP      F10=Unicode  F11=Hold    F12=Alloc
```

The following fields have been added to this screen:

Encoding

The encoding scheme that is used for the data transfer.

Charset Input Error & Replace Count

For transfers using Unicode encoding scheme, specifies the appropriate action when the input file contains data that is not consistent with the specified input character set. The replace count is the number of characters for which the action was taken. For transfers on z/OS systems, the count is the number of data buffers for which the action was taken.

Charset Convert Error & Replace Count

For transfers using Unicode encoding scheme, specifies the action when the input file contains characters that cannot be converted. The characters are not included within the output character sets character repertoire. The replace count is the number of characters for which the action was taken. For transfers on z/OS systems, the count is the number of data buffers for which the action was taken.

Charset

Specifies the character set of the data.

Rec Delimiters

Specifies the encoding scheme for the character set and a set of possible delimiters to use for file processing. This parameter applies only to USS files on z/OS and files on the Linux/Unix/Windows (LUW) platforms.

New and Changed Messages

This section describes the new and changed messages to support this enhancement.

New messages for Unicode transfers are XCOMM0441I, XCOMKM0442I, XCOMM0898I, XCOMM0899I and XCOMM0900E.

0441I

ccccccc CHARSET= xxxxxxxxxxxxxxx

Reason:

When listing the destination member, the default Local and Remote character set information is displayed.

Action:

No action is required.

0442I

ccccccc DELIMITERS= xxxxxxxxxxxxxxx

Reason:

When listing the destination member, the default Local and Remote Encoding and Delimiter information is displayed.

Action:

No action is required.

0899I

UNICODE CONVERSION DETECTED xxxxxxxxxxxxx CHARACTERS IN THE SOURCE DATA

Reason:

When performing character set conversion, data in the source file could not be converted.

When UNCONVERTABLE CHARACTERS is displayed, the source data contains one of the following issues: a character that does not have an equivalent character in the target code page or the substitution character that is defined for the conversion table.

When MALFORMED CHARACTERS is displayed, the source data contains byte strings that do not represent a valid character in the source code page.

Action:

No action is required.

When UNCONVERTABLE CHARACTERS is displayed, If MBCS_INPUTERROR is specified as FAIL, the transfer is terminated. Otherwise the replacement character defined in the conversion table IBM Unicode Services uses is used to replace the character in the target data.

When MALFORMED CHARACTERS is displayed, If MBCS_CONVERROR is specified as FAIL, the transfer is terminated. Otherwise the replacement character defined in the conversion table IBM Unicode Services uses is used to replace the malformed character in the target data.

0900E

UNICODE CONVERSION ERROR - RC=XX REASON=XX – TRANSFER TERMINATED

Reason:

An error condition occurred in IBM Unicode Services performing character conversion.

Action:

Determine the cause of the problem by using the return code and reason code. These codes are documented in the IBM z/OS Unicode Services User Guide and Reference manual.

0901E

PACK=LENGTH HAS BEEN SET FOR A UNICODE TRANSFER

Reason:

When processing a transfer for data in Unicode format, PACK=LENGTH is forced to improve CPU performance for character set conversion processing.

Action:

No action is required.

0902E

A VALID DELIMITER WAS NOT SPECIFIED FOR PATH=xxxxxxxxxxxxxxxxxxxxxxxxxxxx

Reason:

A valid delimiter for the end of the record was not specified for the USS file, preventing the Unicode transfer from completing.

Action:

Rerun the transfer specifying a valid delimiter.

0903E

BPX xxxxxxx ERROR xxxxxxxxx
PATH=xxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxxx xxxxxxxxxxxxxxxxxxxxxxx

Reason:

The specified BPX function for a Unicode(CODE=UTF8 or CODE=UTF16) transfer has failed processing a USS file. The file cannot be processed.

Action:

No action is required.

0907E

XXXXXXXXXXXXXXX MUTUALLY EXCLUSIVE DELIMITER OPTIONS SPECIFIED

Reason:

The delimiter options for either the LOCAL_DELIM or REMOTE_DELIM parameters specify options which are mutually exclusive.

Action:

Modify the delimiter options to remove any mutually exclusive options, which are defined in the CA XCOM Data Transport Users Guide under the LOCAL_DELIM and REMOTE_DELIM SYSIN01 parameters.

0932E

TRUNCATION IS NOT SUPPORTED FOR UNICODE TRANSFERS

Reason:

Using truncation for Unicode data can result in data loss or corruption if truncation occurs in the middle of a MBCS character.

Action:

Initiate the transfer without using truncation. Insure that a large enough logical record length (LRECL) is specified that can hold Unicode data which may have a larger LRECL than the source data.

Chapter 7: XCOMRRDS Transfer Queue Startup Option

A new parameter has been added to provide greater control of the CA XCOM Data Transport server startup.

This section contains the following topics:

New Parameter

This section describes the new PARM and CONFIG parameter.

QSTART

This parameter specifies the handling of the XCOMRRDS transfer queue during initialization of the CA XCOM Data Transport server.

WARM

This option performs a WARM start of the server in normal mode, as for previous releases. The XCOMRRDS transfer queue is read and requests are queued for pending work in the XCOMRRDS data set.

COLD

This option performs a COLD start of the server. It removes all pending work from the RRDS. (This is roughly the equivalent of deleting and defining the XCOMRRDS data set, with the exception that the next transfer request number is not reset to 1000.)

HOLD

This option performs a HOLD start of the server. It reads the XCOMRRDS transfer queue data set and builds requests for all pending work. The difference between HOLD and WARM is that all LOCALLY initiated transfer requests are placed in a HOLD status.

Default: WARM

New Message

This section describes the new message.

0479I

TRANSFER QUEUE BEING xxxx STARTED

Reason:

The CA XCOM Data Transport server is being started with the xxxx parameter for the QSTART option. The following explains the meaning of each possible QSTART option.

WARM: The XCOMRRDS transfer queue data set is being read sequentially, and all transfer requests that were pending when the server was last shutdown will be queued for execution with the same status values at the time the server was terminated.

COLD: All XCOMRRDS transfer request entries in the queue will be deleted, regardless of status. This has the same basic effect as reinitializing the XCOMRRDS data set.

HOLD: The XCOMRRDS transfer queue data set is being read sequentially, and all LOCALLY initiated transfer requests that were pending when the server was last shutdown will be placed in the server queue with a HOLD status.

Action:

None

Appendix A: Deprecated or Obsolete Features

The following features have been deprecated in the r11.6 release of XCOM.

This section contains the following topics:

Default Options Table Deprecated

The Default Options Table is deprecated and replaced by TYPE=CONFIG members in this release. The migration from Default Options Table to TYPE=CONFIG member is simplified by an automatic conversion process the first time a Default Options Table is loaded by a server, admin server, or batch job. In addition, existing batch JCL with the DFLTAB parameter specified interprets the DFLTAB parameter as the CONFIG parameter, without failing the job.

XCOMPLEX Environment Deprecated

The XCOMPLEX deprecated environments include:

- ADMIN server
- TYPE=ADMIN XCONCNTL destination member

The PLEXQ facility introduced in r11.6 is used for a replacement. A PLEXQ environment has been introduced in servers which are also part of an XCOMPLEX. The two coexist so that migration to the PLEXQ is accomplished more easily.

XCOM CICS Interface Deprecated

The CICS interface panels for XCOM are deprecated. The ISPF interface for XCOM is used instead.

TYPE=USER Destination Member Obsolete

The TYPE=USER XCOMCNTL destination member and associated parameters are now obsolete. This destination type used “USER” for a remote system parameter and was historically used for the Windows 3.1 environment and is no longer supported.

Obsolete features include:

- USER batch parameter
- “Login User Name” on ISPF panels
- LOGNREQ system parameter
- TYPE=USER Destination Member

Version 1 Transaction Programs Obsolete

The DFLTVER parameter is now obsolete. Version 1 Transaction Programs for performing CA XCOM Data Transport file transfers have been removed from the product and are no longer available for use. As a result, any value that is specified for the DFLTVER parameter in the #DFLTAB macro generation or in a DEST member is ignored.

Messages are issued indicating that the parameter is no longer supported.

Incoming transfer requests are processed using the current Transaction Program code which can process transfer requests appropriately without regard for the version being requested.

Japanese Character Set Parameters Obsolete

The Japanese Character Set parameters JPNCNTL, JPNDATA, JPNDBL and JPNSHFT, that were used for transfers to the CA XCOM Data Transport for PC-DOS and CA XCOM Data Transport for OS/2 products, have been deprecated. As a result, any value that is specified for any of these parameters in the SYSIN01 file is ignored.

Messages are issued indicating that the parameters are no longer supported.

The Japanese Character Set Support Screen has been removed from the CA XCOM Data Transport ISPF Interface.

Japanese Panels have been removed

The Japanese panels have been removed.

Appendix B: Acknowledgements

This appendix provides acknowledgements for third-party software used with CA XCOM Data Transport for z/OS.

This section contains the following topics:

OpenSSL 0.9.8f

This product includes software developed by the OpenSSL Project for use in the OpenSSL Toolkit (http://www.openssl.org/). This product includes software written by Tim Hudson (tjh@cryptsoft.com).

OpenSSL 0.9.8f is distributed in accordance with the following terms:

```
/* ====================================================================
 * Copyright (c) 1998-2007 The OpenSSL Project.  All rights reserved.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 *
 * 1. Redistributions of source code must retain the above copyright
 *    notice, this list of conditions and the following disclaimer.
 *
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in
 *    the documentation and/or other materials provided with the
 *    distribution.
 *
 * 3. All advertising materials mentioning features or use of this
 *    software must display the following acknowledgment:
 *    "This product includes software developed by the OpenSSL Project
 *    for use in the OpenSSL Toolkit. (http://www.openssl.org/)"
 *
 * 4. The names "OpenSSL Toolkit" and "OpenSSL Project" must not be used to
 *    endorse or promote products derived from this software without
 *    prior written permission. For written permission, please contact
 *    openssl-core@openssl.org.
 *
 * 5. Products derived from this software may not be called "OpenSSL"
 *    nor may "OpenSSL" appear in their names without prior written
 *    permission of the OpenSSL Project.
 *
 * 6. Redistributions of any form whatsoever must retain the following
 *    acknowledgment:
 *    "This product includes software developed by the OpenSSL Project
 *    for use in the OpenSSL Toolkit (http://www.openssl.org/)"
 *
 * THIS SOFTWARE IS PROVIDED BY THE OpenSSL PROJECT ``AS IS|&"&| AND ANY
 * EXPRESSED OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE
 * IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR
 * PURPOSE ARE DISCLAIMED.  IN NO EVENT SHALL THE OpenSSL PROJECT OR
 * ITS CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL,
 * SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT
 * NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES;
 * LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION)
 * HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT,
 * STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE)
 * ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED
 * OF THE POSSIBILITY OF SUCH DAMAGE.
 * ====================================================================
```

```
*
* This product includes cryptographic software written by Eric Young
* (eay@cryptsoft.com).  This product includes software written by Tim
* Hudson (tjh@cryptsoft.com).
*
*/

Original SSLeay License
-----------------------

/* Copyright (C) 1995-1998 Eric Young (eay@cryptsoft.com)
 * All rights reserved.
 *
 * This package is an SSL implementation written
 * by Eric Young (eay@cryptsoft.com).
 * The implementation was written so as to conform with Netscapes SSL.
 *
 * This library is free for commercial and non-commercial use as long as
 * the following conditions are aheared to.  The following conditions
 * apply to all code found in this distribution, be it the RC4, RSA,
 * lhash, DES, etc., code; not just the SSL code.  The SSL documentation
 * included with this distribution is covered by the same copyright terms
 * except that the holder is Tim Hudson (tjh@cryptsoft.com).
 *
 * Copyright remains Eric Young's, and as such any Copyright notices in
 * the code are not to be removed.
 * If this package is used in a product, Eric Young should be given attribution
 * as the author of the parts of the library used.
 * This can be in the form of a textual message at program startup or
 * in documentation (online or textual) provided with the package.
 *
 * Redistribution and use in source and binary forms, with or without
 * modification, are permitted provided that the following conditions
 * are met:
 * 1. Redistributions of source code must retain the copyright
 *    notice, this list of conditions and the following disclaimer.
 * 2. Redistributions in binary form must reproduce the above copyright
 *    notice, this list of conditions and the following disclaimer in the
 *    documentation and/or other materials provided with the distribution.
 * 3. All advertising materials mentioning features or use of this software
 *    must display the following acknowledgement:
 *    "This product includes cryptographic software written by
 *     Eric Young (eay@cryptsoft.com)"
 *    The word 'cryptographic' can be left out if the rouines from the library
 *    being used are not cryptographic related :-).
 * 4. If you include any Windows specific code (or a derivative thereof) from
 *    the apps directory (application code) you must include an acknowledgement:
 *    "This product includes software written by Tim Hudson (tjh@cryptsoft.com)"
 *
```

ANY EXPRESS OR IMPLIED WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE
ARE DISCLAIMED. IN NO EVENT SHALL THE AUTHOR OR CONTRIBUTORS BE LIABLE FOR ANY DIRECT, INDIRECT, INCIDENTAL, SPECIAL, EXEMPLARY, OR CONSEQUENTIAL DAMAGES (INCLUDING, BUT NOT LIMITED TO, PROCUREMENT OF SUBSTITUTE GOODS OR SERVICES; LOSS OF USE, DATA, OR PROFITS; OR BUSINESS INTERRUPTION) HOWEVER CAUSED AND ON ANY THEORY OF LIABILITY, WHETHER IN CONTRACT, STRICT LIABILITY, OR TORT (INCLUDING NEGLIGENCE OR OTHERWISE) ARISING IN ANY WAY OUT OF THE USE OF THIS SOFTWARE, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

The license and distribution terms for any publically available version or derivative of this code cannot be changed. i.e. this code cannot simply be copied and put under another distrubution license
[including the GNU Public License.]

The reason behind this being stated in this direct manner is past experience in code simply being copied and the attribution removed from it and then being distributed as part of other packages. This implementation was a non-trivial and unpaid effort.

Zlib 1.0.2

This product contains Zlib, which is distributed in accordance with the following license agreement:

Copyright (C) 1995-2010 Jean-loup Gailly and Mark Adler

This software is provided 'as-is', without any express or implied

warranty. In no event will the authors be held liable for any damages

arising from the use of this software.

Permission is granted to anyone to use this software for any purpose,

including commercial applications, and to alter it and redistribute it

freely, subject to the following restrictions:

1. The origin of this software must not be misrepresented; you must not

claim that you wrote the original software. If you use this software

in a product, an acknowledgment in the product documentation would be

appreciated but is not required.

2. Altered source versions must be plainly marked as such, and must not be

misrepresented as being the original software.

3. This notice may not be removed or altered from any source distribution.

Jean-loup Gailly

Mark Adler

Index

M