XCF Option Administration Guide

Administration

Version 6.5.1

March 2023

ADABAS & NATURAL

This document applies to Entire Net-Work XCF Option Version 6.5.1 and all subsequent releases.

Specifications contained herein are subject to change and these changes will be reported in subsequent release notes or new editions.

Document ID: WXF-DOC-651-20230324

Table of Contents

Preface

This document provides information for administrators responsible for configuring and running the Entire Net-Work XCF line driver once Entire Net-Work is installed.

Note: The XCF line driver is provided in the Software AG product option called the Entire Net-Work XCF Option (product code WXF), which is an add-on to the Entire Net-Work product and must be ordered separately.

The Entire Net-Work XCF line driver can be used in a sysplex to provide high performance, transparent communications between client/server cooperating processes (Adabas and Natural) and server-to-server communication applications (Adabas Cluster Services) that reside on different central processors in the sysplex.

The XCF line driver documentation is organized as follows:

Entire Net-Work XCF Option Release Notes	Describes changes made for this release of the Entire Net-Work XCF Option.
XCF Overview	Provides an overview of IBM's Cross-System Coupling Facility (XCF).
Preparing the XCF Environment	Describes what you need to do to prepare the XCF environment.
Entire Net-Work XCF Option Installation Procedure	Describes the installation steps for the (the XCF line driver)
XCF DRIVER Statement	Describes the syntax and parameters of the XCF DRIVER statement.
XCF LINK Statement	Describes the syntax and parameters of the XCF LINK statement.
XCF Operator Commands	Lists and describes the Entire Net-Work XCF Option operator commands.

1 Conventions

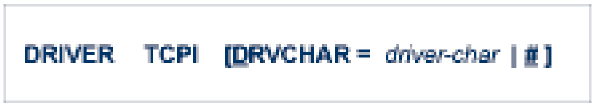
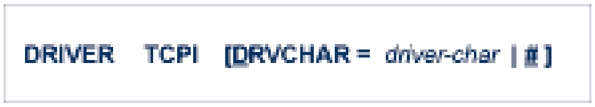
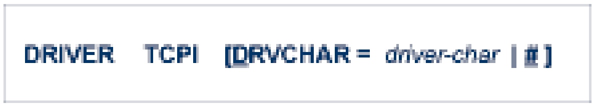
Notation "`vr` SP `s`", `vrs`, or `vr`: When used in this documentation, the notation "`vr` SP `s`", `vrs`, or `vr` stands for the relevant version, release, and system maintenance level numbers. For further information on product versions, see *version* in the *Glossary*.

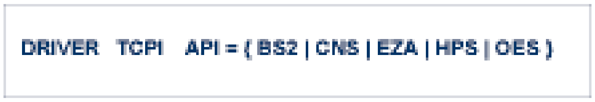
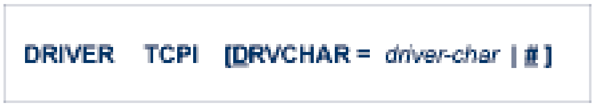
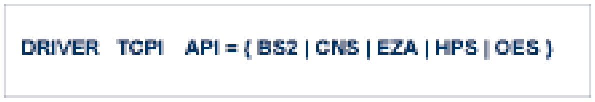
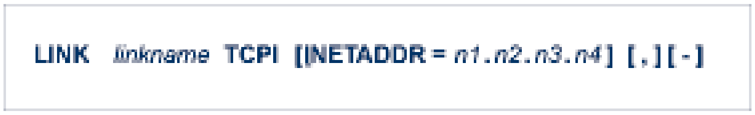
This document covers the following topics:

- Syntax Conventions
- Syntax Rules

Syntax Conventions

The following table describes the conventions used in syntax diagrams of Entire Net-Work statements.

Convention	Description	Example
uppercase, bold	Syntax elements appearing in uppercase and bold font are keywords. When specified, these keywords must be entered exactly as shown.	 **DRIVER TCPI [DRVCHAR =** *driver-char* | **#]** The syntax elements DRIVER, TCPI, and DRVCHAR are Entire Net-Work keywords.
lowercase, italic, normal font	Syntax elements appearing in lowercase and normal, italic font identify items that you must supply.	 **DRIVER TCPI [DRVCHAR =** *driver-char* | **#]** The syntax element *`driver-char`* identifies and describes the kind of value you must supply. In this instance, you must supply the special character used to designate that an operator command is directed to the TCP/IP line driver, rather than to a specific link.
underlining	Underlining is used for two purposes: 1. To identify default values, wherever appropriate. Otherwise, the defaults are explained in the accompanying parameter descriptions. 2. To identify the short form of a keyword.	 **DRIVER TCPI [DRVCHAR =** *driver-char* | **#]** In the example above, # is the default that will be used for the DRVCHAR parameter if no other record buffer length is specified. Also in the example above, the short version of the DRVCHAR parameter is D.

Convention	Description	Example
vertical bars (|)	Vertical bars are used to separate mutually exclusive choices. **Note:** In more complex syntax involving the use of large brackets or braces, mutually exclusive choices are stacked instead.	 DRIVER TCPI API = { BS2 | CNS | EZA | HPS | OES } In the example above, you must select BS2, CNS, EZA, HPS, or OES for the API parameter. There are no defaults.
brackets ([ ])	Brackets are used to identify optional elements. When multiple elements are stacked or separated by vertical bars within brackets, only one of the elements may be supplied.	 DRIVER TCPI [DRVCHAR = *driver-char* | #] In this example, the DRVCHAR parameter is optional.
braces ({ })	Braces are used to identify required elements. When multiple elements are stacked or separated by vertical bars within braces, one and only one of the elements must be supplied.	 DRIVER TCPI API = { BS2 | CNS | EZA | HPS | OES } In this example, one of the following values is required for the API parameter: BS2, CNS, EZA, HPS, or OES.
other punctuation and symbols	All other punctuation and symbols must be entered exactly as shown.	 LINK *linkname* TCPI [NETADDR = *n1.n2.n3.n4*] [,] [-] In this example, the periods must be specified in the IP address. In addition, options must be separated by commas and dashes should be used as needed to indicate that parameter settings continue on the next line.

Syntax Rules

The following rules apply when specifying Entire Net-Work parameter statements:

- Each Entire Net-Work parameter statement occupies positions 1 - 72 of at least one line.
- The statement type (NODE, LINK, TRANSDEF, or DRIVER) must be specified as the first non-blank item on the statement.
- The node name, driver name, translation definition function, or link name follows the statement type, separated by at least one blank (space).

- Keyword parameters may be specified following either the node name on NODE statements or the driver name on DRIVER and LINK statements. Keyword parameters are separated from their arguments by an equal (=) sign, and from other keyword parameters by at least one blank (space) or a comma (,).
- When the acceptable values for a parameter are Y and N (yes and no), any other value is treated as an N, unless there is a documented default, and processing continues without any warning.
- When the acceptable values for a parameter fall within a range (e.g., 1 - 2147483647) and a value outside the range is specified, the value is automatically reset to the maximum value within the range, unless documented otherwise for the parameter. Processing continues without any warning.
- A statement can be continued beginning in any column of the next line by specifying a dash (-) as the last nonblank character in any column of the current line, before column 73.
- Comment lines begin with an asterisk (*) in position 1 and can be inserted anywhere in the statement sequence.
- Some keywords may require a list of subparameters separated by commas; the list must be enclosed in parentheses ( ) unless only the first subparameter is to be entered. Omitted ("defaulted") subparameters must be represented by placeholder commas if subsequent parameters are to be entered. The following are examples of correct subparameter strings:

```
KEYWORD=(value1,value2,value3)
KEYWORD=(value1,,value3)
KEYWORD=(,,value3)
KEYWORD=(,value2)
KEYWORD=value1
```

- Hexadecimal keyword values can be entered by prefixing the value with an "X". For example:

```
LINK . . . ADJID=X0064, . . .
```

2 About this Documentation

Document Conventions

Convention	Description
Bold	Identifies elements on a screen.
`Monospace font`	Identifies service names and locations in the format *folder.subfolder.service*, APIs, Java classes, methods, properties.
Italic	Identifies: Variables for which you must supply values specific to your own situation or environment. New terms the first time they occur in the text. References to other documentation sources.
`Monospace font`	Identifies: Text you must type in. Messages displayed by the system. Program code.
{ }	Indicates a set of choices from which you must choose one. Type only the information inside the curly braces. Do not type the { } symbols.
|	Separates two mutually exclusive choices in a syntax line. Type one of these choices. Do not type the | symbol.
[ ]	Indicates one or more options. Type only the information inside the square brackets. Do not type the [ ] symbols.
...	Indicates that you can type multiple options of the same type. Type only the information. Do not type the ellipsis (...).

Online Information and Support

Product Documentation

You can find the product documentation on our documentation website at **https://documentation.softwareag.com**.

In addition, you can also access the cloud product documentation via **https://www.softwareag.cloud**. Navigate to the desired product and then, depending on your solution, go to “Developer Center”, “User Center” or “Documentation”.

Product Training

You can find helpful product training material on our Learning Portal at **https://knowledge.softwareag.com**.

Tech Community

You can collaborate with Software AG experts on our Tech Community website at **https://tech-community.softwareag.com**. From here you can, for example:

- Browse through our vast knowledge base.
- Ask questions and find answers in our discussion forums.
- Get the latest Software AG news and announcements.
- Explore our communities.
- Go to our public GitHub and Docker repositories at **https://github.com/softwareag** and **https://hub.docker.com/publishers/softwareag** and discover additional Software AG resources.

Product Support

Support for Software AG products is provided to licensed customers via our Empower Portal at **https://empower.softwareag.com**. Many services on this portal require that you have an account. If you do not yet have one, you can request it at **https://empower.softwareag.com/register**. Once you have an account, you can, for example:

- Download products, updates and fixes.
- Search the Knowledge Center for technical information and tips.
- Subscribe to early warnings and critical alerts.
- Open and update support incidents.
- Add product feature requests.

Data Protection

Software AG products provide functionality with respect to processing of personal data according to the EU General Data Protection Regulation (GDPR). Where applicable, appropriate steps are documented in the respective administration documentation.

3 Entire Net-Work XCF Option Release Notes

Effective with version 6.3 SP2, zEnterprise Data Compression (zEDC) is supported on z/OS systems. This support comes in the form of two new LINK parameters for the XCF line driver: ZEDC and ZEDCLOG.

The following table summarizes the new Entire Net-Work XCF Option DRIVER and LINK parameters introduced in Entire Net-Work 6.3:

Parameter	DRIVER or LINK Statement?	New or Changed?	Description	Introduced in Release
EXHS	DRIVER and LINK	New	This new parameter indicates whether extended handshakes should be used. Specify this parameter only as advised by Software AG; it was created to provide compatibility between the Entire Net-Work XCF Option 6.3 releases and the 6.1 SP2 and 6.2 SP2 releases. Zaps are required to support this functionality. .	6.3 SP1
RCVBFNUM	DRIVER	New	This optional new parameter allows you to specify the number of entries in the receive buffer table. Zaps are provided to support this functionality.	6.3 SP2
ZEDC	LINK	New	This new parameter indicates whether zEnterprise Data Compression (zEDC) compression can occur for the link.	6.3 SP2
ZEDCLOG	LINK	New	This new parameter indicates what level of trace data will be logged for zEDC compression processing.	6.3 SP2

For more information, read *XCF LINK Statement*(elsewhere in this guide).

4 XCF Overview

IBM's Cross-System Coupling Facility (XCF) allows authorized applications on one system to communicate with applications on the same system or on other systems. XCF transfers data and status information between members of a group that resides on one or more OS/390 or z/OS systems in a *sysplex* (SYStems comPLEX).

A *sysplex* is a set of software services and hardware components that allow OS/390 or z/OS systems to communicate and cooperate with each other to process customer work. For certain kinds of work, the sysplex provides parallel processing and improved data sharing.

A *member* is a specific function (one or more modules/routines) of a multisystem application that is defined to XCF and assigned to a group by the multisystem application. A member resides on one OS/390 or z/OS system in the sysplex and can use XCF services to communicate (send and receive data) with other members of the same group. Each Entire Net-Work node running the XCF line driver is identified as a different member in a group specifically set up for Entire Net-Work connectivity.

High-speed caching, list processing, and locking functions are provided by the sysplex hardware components. The *coupling facility* is a microprocessor unit that allows up to 32 OS/390 or z/OS systems to share data.

OS/390 or z/OS systems are directly connected to the coupling facilities through high bandwidth, high speed fiber optic links called *coupling facility channels*. Coupling facility channels provide a fast and efficient means for systems in the sysplex to access data in the coupling facility.

The coupling facility channel can use 50/125 multimode fiber at 531 megabits per second (Mbps), supporting a maximum distance of approximately one kilometer between the coupling facility and the processor.

5 Preparing the XCF Environment

Each OS/390 or z/OS system that needs access to the sysplex coupling facility must be connected with at least one coupling facility channel, whether the system runs on the same processor as the Coupling Facility Central Processor (CFCP) or on a separate processor.

Channel Path Attachments

As part of the I/O configuration definitions for each coupling facility channel, you need to define the channel path attachments for all processors and coupling facility channels.

1. Use coupling facility channels to connect the coupling facility to the processors in the sysplex.
2. When defining the coupling facility, you must dedicate a single central processor (CP) to the partition that runs the CFCP.
3. The CFCP does not support dynamic I/O reconfiguration or PR/SM LPAR reconfiguration. To activate reconfiguration changes, you must power-on reset (POR) the processor that runs the CFCP.
4. Define channel paths for both the coupling facility channel attachment to the processor and the coupling facility channel attachment to the coupling facility. For each system in the sysplex, you can use the HCD panels to define the I/O configuration data set (IOCDS) that must include definitions for the coupling facility channel paths.

COUPLExx PARMLIB Member

You must specify a COUPLE*xx* PARMLIB member in SYS1.PARMLIB for each system in the sysplex. The COUPLE*xx* member defines the following values:

- Sysplex name, sysplex couple data set names, and other couple data set names.
- Failure detection interval, operator notification interval, and cleanup interval.
- Default message buffer space, default message size for a signaling path, and local message buffer space.
- Transport classes.
- Outbound and inbound signaling paths.
- Default retry limit.

References

For additional information, refer to IBM's Sysplex documentation.

6 Entire Net-Work XCF Option Installation Procedure

The Entire Net-Work XCF Option currently supports the z/OS operating systems only.

If Entire Net-Work is not already installed, see the section *Installation Overview* in the *Entire Net-Work Installation Guide*. Then perform the following additional steps:

1. Using IEBCOPY, unload the Entire Net-Work XCF Option library from the installation media into your Entire Net-Work load library.

   Refer to the *Software AG Product Delivery Report* for the correct data set sequence numbers and names.
2. The XCF line driver requires APF authorization. To make Entire Net-Work APF-authorized:
   - Ensure that all load libraries in your STEPLIB concatenation have been defined to z/OS as authorized libraries; and
   - Ensure that module ADARUN has been linked with AMODE(31) and AC(1).
3. Add a DRIVER statement that specifies your requirements to the Entire Net-Work parameter data set. The DRIVER statement is required. For more information, read ***XCF DRIVER Statement*** elsewhere in this guide.
4. Optionally, add one or more LINK statements. LINK statements describe Entire Net-Work partner nodes, and are currently not required. For more information, read ***XCF LINK Statement*** elsewhere in this guide.
5. Define the z/OS-related XCF parameters:
   - Class and Group for Entire Net-Work.
   - Entire Net-Work list structures.

   For more information, read ***XCF Line Driver Values*** elsewhere in this guide.
6. Start Entire Net-Work (on multiple nodes) and run verification tests. For example:
   - Test your applications running across Entire Net-Work. Run one application at a time and verify the results.
   - Run a load test through the network (that is, multiple users on each node accessing data on the partner node).

XCF Line Driver Values

Software AG recommends that you add the values used by the Entire Net-Work XCF line driver to the COUPLE*xx* member in SYS1.PARMLIB, as follows:

- Add a CLASSDEF statement with CLASS(ENTIRE) and GROUP(NETWORK) specified.
- Add a PATHIN statement with STRNAME(IXC_NETWORK) specified.
- Add a PATHOUT statement with STRNAME(IXC_NETWORK) and CLASS(ENTIRE) specified.

For example:

```
COUPLE SYSPLEX(USRSPLX1)
       PCOUPLE(SYS1.SYSPLEX.USRSPLX1.CDS01)
           ACOUPLE(SYS1.SYSPLEX.USRSPLX1.CDS02)
           INTERVAL(120)

CLASSDEF
           CLASS(ENTIRE)
           CLASSLEN(8192)
           GROUP(NETWORK)
           MAXMSG(4096)
```

```
LOCALMSG
           MAXMSG(4096)
           CLASS(ENTIRE)
PATHIN STRNAME(IXC_DEFAULT,IXC_NETWORK)
PATHOUT STRNAME(IXC_DEFAULT)
PATHOUT STRNAME(IXC_NETWORK) CLASS(ENTIRE)
```

Note: No changes to the COUPLE*xx* member are required. If values are not specified, the system defaults will be used.

You will need to run a job similar to the following example in order to define to XCF a list structure for Entire Net-Work:

```
//STEP20   EXEC PGM=IXCMIAPU
//SYSPRINT DD   SYSOUT=X
//SYSABEND DD   SYSOUT=X
//SYSIN    DD   *

     DATA TYPE(CFRM) REPORT(YES)

     DEFINE POLICY NAME(POLICY1) REPLACE(YES)
          CF NAME(CF)
              TYPE(009674)
                   MFG(IBM)
                   PLANT(02)
                   SEQUENCE
                   PARTITION(1)
                   CPCID(00)
                   DUMPSPACE(2000)
          STRUCTURE NAME(IXC_DEFAULT)
                   SIZE(1000)
                   PREFLIST(CF)
          STRUCTURE NAME(IXC_NETWORK)
```

```
                   SIZE(8192)
                   PREFLIST(CF) ↩
```

Contents of the Release Media

The following table describes most of the libraries included on the release (installation) media. Once you have unloaded the libraries from the media, you can change these names as required by your site, but the following lists the names that are delivered when you purchase Entire Net-Work XCF Option.

Note: The complete list of libraries provided with Entire Net-Work can be found in the full Entire Net-Work documentation.

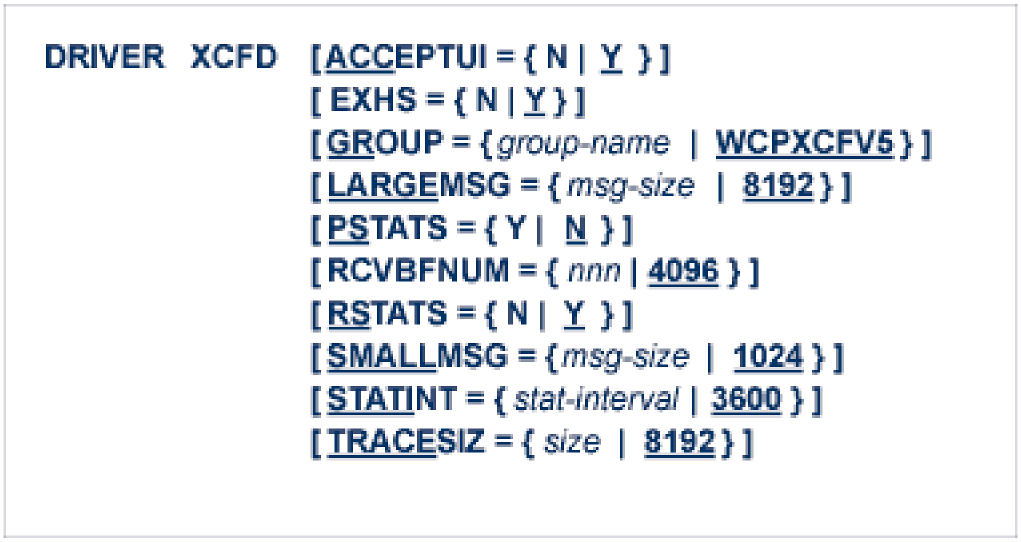
Library Name	Description
WXF*vrs*.LOAD	The z/OS load library for Entire Net-Work XCF Option. The *vrs* in the library name represents the *version* of Entire Net-Work XCF Option.

7 XCF DRIVER Statement

The XCF DRIVER statement and its parameters are used to activate and define the characteristics of the local IBM mainframe node. The access method name "XCFD" instructs Entire Net-Work to load the line driver module NETXCF.

XCF DRIVER Statement Format

The XCF DRIVER statement has the following format:



```
DRIVER  XCFD  [ACCEPTUI = { N | Y }]
              [ EXHS = { N | Y } ]
              [ GROUP = {group-name  |  WCPXCFV5 } ]
              [ LARGEMSG = { msg-size  |  8192 } ]
              [ PSTATS = { Y |  N  } ]
              [ RCVBFNUM = { nnn | 4096 } ]
              [ RSTATS = { N |  Y  } ]
              [ SMALLMSG = { msg-size  |  1024 } ]
              [ STATINT = { stat-interval | 3600 } ]
              [ TRACESIZ = { size  |  8192 } ]
```

For more information about syntax conventions and rules used in this section, read *Conventions*.

Modifying the DRIVER Statement Parameters

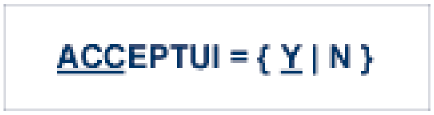
The DRIVER statement parameters are read from a sequential file during system startup, and can be modified after startup using the ALTER operator command. Some parameters can be modified when the line driver is open or closed. Others can be modified only when the line driver is closed. For more information about the CLOSE command, read *Entire Net-Work Operator Commands* in the *Entire Net-Work Reference Guide*. For more information about the ALTER command, read *XCF Operator Commands*elsewhere in this guide. The open/closed requirement for each parameter is included in its description.

DRIVER Statement Parameters

This section describes all of the parameters that can be used for the XCF DRIVER statement.

For more information about syntax conventions and rules used in this section, read *Conventions*.

ACCEPTUI Parameter



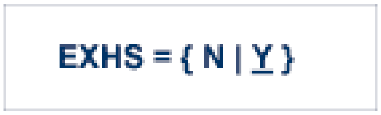
ACCEPTUI = { Y | N }

This optional parameter determines whether the line driver will accept connections from systems that have not been previously defined with LINK statements. The ACCEPTUI parameter can be modified when the line driver is open or closed.

Valid values are "Y" (Yes) or "N" (No).

- If "Y" is specified, Entire Net-Work will accept connection requests from an undefined system and the required control blocks are built dynamically. Normal "handshaking" procedures with the new connections are performed.
- If "N" is specified, Entire Net-Work will reject incoming requests from unknown source nodes.

EXHS Parameter



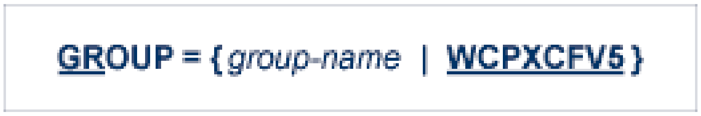
EXHS = { N | Y }

This optional parameter controls whether extended handshakes are used. In most environments this parameter should not be specified. Software AG recommends that EXHS=Y not be used; allow EXHS to default to "Y" instead. EXHS=N should be specified in *only* the following situations:

- If Entire Net-Work 6.3 SP2 nodes need to make XCF connections to nodes that are still version 6.1 SP2, and the version 6.1 SP2 nodes *do not* have zap WM612033 applied, specify EXHS=N for the version 6.3 SP2 XCF line driver.
- If Entire Net-Work 6.3 SP2 nodes need to make XCF connections to nodes that are still version 6.2 SP2, and the version 6.2 SP2 nodes *have* optional zap WY622003 applied, specify EXHS=N for the 6.3 SP2 XCF line driver.

After all nodes have been upgraded to Entire Net-Work 6.3 SP2, the EXHS parameter should be removed node-by-node as scheduling allows. The parameter does not have to be removed from all nodes at once.

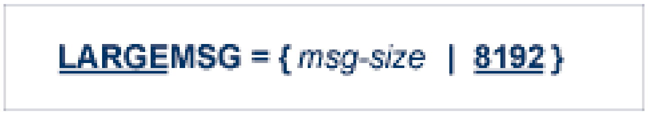
GROUP Parameter



GROUP = { *group-name* | WCPXCFV5 }

All Entire Net-Work nodes intending to form part of the same network environment on the sysplex must use the same unique XCF group name. The group name must be the same on all Entire Net-Work nodes; it must also be different from the name of any other product or system using XCF. Results are unpredictable if Entire Net-Work nodes are assigned to the same group as other types of systems such as DB2 or CICS. Multiple XCF group names must be defined if you wish to support multiple isolated/separated network environments within one sysplex. The GROUP parameter specifies the Entire Net-Work group name, which may be defined in the COUPLE*xx* member in SYS1.PARMLIB (see Step 5 of the *Entire Net-Work XCF Option Installation Procedure*). The default group name is WCPXCFV5. The GROUP parameter can be modified only when the driver is closed.

LARGEMSG Parameter



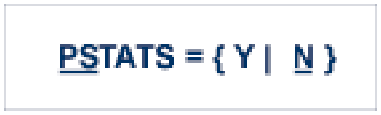
LARGEMSG = { *msg-size* | 8192 }

This optional parameter is used to specify the minimum size of a large message for statistics reporting. Any data messages larger than this size are counted as large messages. Any data messages smaller than this size are counted as medium or small messages, depending on the value specified by the SMALLMSG parameter.

For example, using the default values for SMALLMSG and LARGEMSG:

- Data messages between 1 and 1024 bytes long are reported as small messages.
- Data messages between 1025 and 8192 bytes long are reported as medium messages.
- Data messages larger than 8192 bytes long are reported as large messages.

PSTATS Parameter



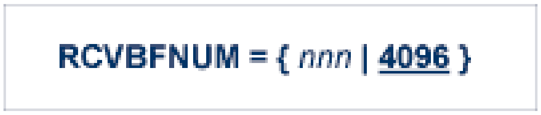
PSTATS = { Y | N }

This optional parameter determines whether or not statistics are printed. Statistics triggered by the PSTATS parameter are written only to DDPRINT.

A value of "Y" indicates that statistics should be printed at regular intervals; a value of "N" indicates that the statistics should not be printed. The default is "N".

This parameter is only used to set the default for all links.

RCVBFNUM Parameter



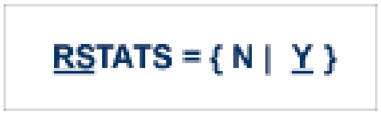
RCVBFNUM = { *nnn* | 4096 }

This optional parameter allows you to specify the number of entries in the receive buffer table. The minimum value you can specify is 100, the maximum is 99999. The default is 4096.

Values outside the valid range are set to the respective minimum or maximum with no error message. Each entry is 16 bytes. For example, RCVBFNUM=8192 would create a table with 8192 entries, a total size of 131072 bytes.

Note: In almost all cases the default table size is adequate. We recommend that you do not alter the RCVBFNUM parameter setting unless directed to do so by Software AG.

RSTATS Parameter



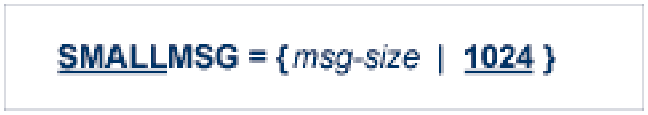
RSTATS = { N | Y }

This optional parameter determines whether or not statistics are reset.

A value of "Y" indicates that statistics should be rest at regular intervals; a value of "N" indicates that the statistics should not be reset.

This parameter is only used to set the default for all links.

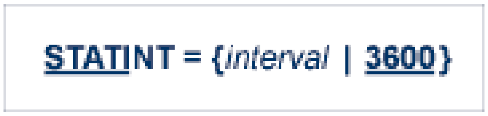
SMALLMSG Parameter



SMALLMSG = {*msg-size* | 1024 }

This optional parameter is used to specify the maximum size of a small message for statistics reporting. Any data messages smaller than or equal this size are counted as small messages. Any data messages larger than this size are counted as either medium or large messages, depending on the value specified by the LARGMSG parameter.

STATINT Parameter



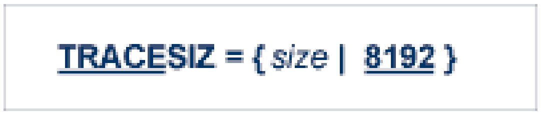
STATINT = {*interval* | 3600}

This optional parameter specifies the amount of time, in seconds, before statistics are automatically printed or reset. Statistics triggered by the STATINT parameter are written only to DDPRINT.

The STATINT parameter can be modified when the line driver is open or closed. Acceptable values range from 0 to 2147483647. The default is 3600.

No statistics are collected or reported for the driver; this parameter is used to set the default for all links.

TRACESIZ Parameter



TRACESIZ = { *size* | 8192 }

This optional parameter specifies the size, in bytes, of the driver-specific trace table.

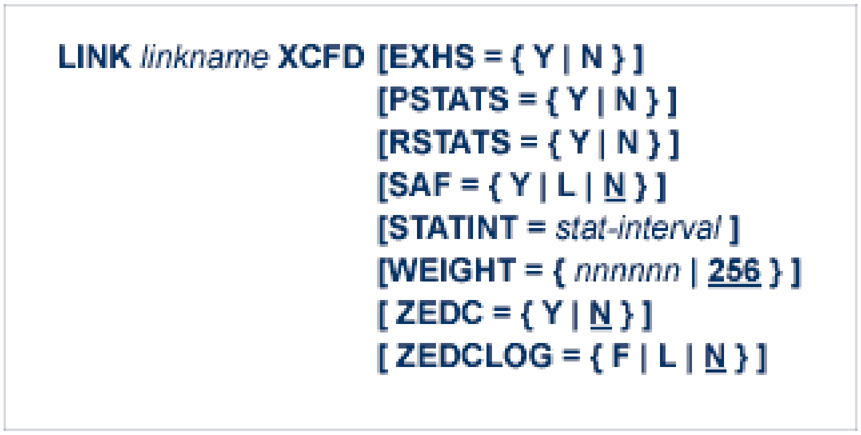
The TRACESIZ parameter can be modified when the driver is open or closed

8 XCF LINK Statement

The LINK statement is not required. During the initialization process, the XCF line driver requests a list of all other members in its group and then logically connects to each member.

Although the LINK statement is optional, it can be added for documentation or security purposes.

Note: If two or more Entire Net-Work nodes are active on the sysplex within the same group, the Entire Net-Work operator command 'D L' can be used to display the list of dynamically added links and parameter-defined links.



```
LINK linkname XCFD [EXHS = { Y | N } ]
                   [PSTATS = { Y | N } ]
                   [RSTATS = { Y | N } ]
                   [SAF = { Y | L | N } ]
                   [STATINT = stat-interval ]
                   [WEIGHT = { nnnnnn | 256 } ]
                   [ ZEDC = { Y | N } ]
                   [ ZEDCLOG = { F | L | N } ]
```

For more information about syntax conventions and rules used in this chapter, read *Conventions*.

Modifying the LINK Statement Parameters

The LINK statement parameters are read from a sequential file during system startup, and can be modified after startup using the ALTER operator command (see the section *XCF Operator Commands*). The parameters can be modified when the link is open or closed.

LINK Statement Parameters

In the following syntax-example diagrams, the underlined portion of the parameter name is the minimum abbreviation of the parameter.

linkname Parameter

```
linkname
```

This required parameter specifies the name by which this link is to be known. It is positional, and must be specified immediately after the LINK keyword and immediately before the driver name (XCFD). The linkname must be unique on the node, and must match the node name of some member in the XCF group; that is, it must match the node name of the member being connected to.

EXHS Parameter

```

EXHS={Y | N}
```

This optional parameter determines whether or not (Y or N) extended handshakes are used on this link. This parameter should be used only when directed by Software AG..

The default value is the value of the EXHS parameter on the ***XCF DRIVER Statement***.

PSTATS Parameter

```

PSTATS={Y | N}
```

This optional parameter determines whether or not (Y or N) statistics are printed automatically at regular intervals. The default value is the value of the PSTATS parameter on the ***XCF DRIVER Statement***.

Statistics triggered by the PSTATS parameters are written only to DDPRINT.

RSTATS Parameter

```

RSTATS={Y | N}
```

This optional parameter determines whether or not (Y or N) statistics are reset automatically at regular intervals. The default value is the value of the RSTATS parameter on the ***XCF DRIVER Statement***.

SAF Parameter

```
SAF={Y | L | N}
```

If SAF=Y or SAF=L is specified, Entire Net-Work will call the SAF interface for all incoming requests on this link; failure to load the Interface is considered a security violation and Entire Net-Work will shut down. If SAF=L, the calls are traced and the output directed to DDPRINT. An error code is transmitted to the user if access to SAF is denied. The SAF parameter can be modified when the link is open or closed. The default value is N (No).

STATINT Parameter

```
STATINT= statinterval
```

This optional parameter specifies the amount of time, in seconds, before statistics are automatically reported or reset. The value may be 0 - 2147483647. The default value is the value of the STATINT parameter on the ***XCF DRIVER Statement***.

Statistics triggered by the STATINT parameters are written only to DDPRINT.

WEIGHT Parameter

```
WEIGHT={n | 256}
```

This parameter specifies the weight of this link with respect to other links going to the same node. If a given target can be reached by more than one path (chain of connected links), the path with the lowest weight is used. Slow or expensive links should be given a higher value than fast or inexpensive links. Values range from 1 to 999999. The default value is 256. The WEIGHT parameter can be modified only when the link is closed.

ZEDC Parameter

```
ZEDC={ Y | N }
```

This parameter indicates whether zEnterprise Data Compression (zEDC) compression can occur for the link. Valid values are "Y" or "N"; "N" is the default. Determination of whether or not zEDC data compression occurs is based on a combination of the settings of this parameter and the ZEDCINIT parameter on the NODE statement, as described in the following table:

LINK ZEDC Parameter Setting	NODE ZEDCINIT Parameter Setting	Result
Y	Y	Outbound buffers for the link are compressed.
Y	N	Outbound buffers are not compressed.
N	Y	Outbound buffers for TCPI links are not compressed, but other outbound buffers might be (depending on the setting of their LINK statement ZEDC parameters).
N	N	Outbound buffers are not compressed.

Note: If the node-to-node handshake indicates that the destination node does not support zEDC data compression, the outbound payload will not be compressed, regardless of any zEDC parameter settings on the NODE statement or any LINK statement.

zEnterprise Data Compression (zEDC) can occur only on z/OS operating systems. Consequently, ZEDC=Y can be specified only on z/OS systems that support zEDC. For complete information on z/OS requirements for zEDC support, refer to IBM documentation regarding *zEnterprise Data Compression (zEDC)*.

When compression occurs it occurs on buffers with sizes greater than the value defined by the NODE statement's ZEDCSZ parameter.

ZEDCLOG Parameter

```
ZEDCLOG={ F | L | N }
```

This optional parameter indicates what level of trace data will be logged for zEDC compression processing. This trace data logging occurs independently of Entire Net-Work's global tracing parameter setting (LOG=YES or LOG=FULL parameter settings on the NODE statement). Valid values are described in the following table:

ZEDCLOG Setting	Result
F	Trace data is logged prior to and after compression and decompression processing. The amount of data logged is equivalent to the length of the data.
L	Trace data is logged prior to and after compression and decompression processing. The amount of data logged is 100 (x'64') bytes.
N	This is the default. No trace data is logged.

Note: The F and L settings of ZEDCLOG should be used sparingly; these settings greatly increase the DDPRINT output size.

The ZEDCLOG parameter, can be modified when a link is open or closed.

Note: If the node-to-node handshake indicates that the destination node does not support zEDC data compression, the outbound payload will not be compressed, regardless of any zEDC parameter settings on the NODE statement or any LINK statement.

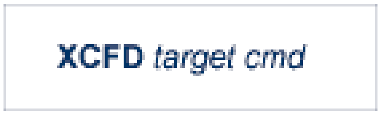
zEnterprise Data Compression (zEDC) can occur only on z/OS operating systems. Consequently, the ZEDCSLOG parameter specification should be made only on z/OS systems that support zEDC. For complete information on z/OS requirements for zEDC support, refer to IBM documentation regarding *zEnterprise Data Compression (zEDC)*.

9 XCF Operator Commands

Entire Net-Work's XCF line driver has the ability to process operator commands that are directed to a specific link, to all links, or directly to the driver.

Operator Command Syntax

The XCF line driver operator commands have the following format:



XCFD *target cmd*

The following table describes this syntax.

Syntax Representation	Description
XCFD	Informs Entire Net-Work that the command is destined for the XCF line driver.
target	A value that informs Entire Net-Work what the target of the command is, as follows: ■ Specify an asterisk (*) if the target is all links. ■ Specify the pound sign (#) if the target is the driver itself. ■ Specify the link name if the target is a specific link.
cmd	The operator command to be carried out.

Examples

The following are examples of operator commands:

```
XCFD * SHOW
```

```
XCFD # STATUS
```

```
XCFD link3 TRACE
```

Driver Commands

The Entire Net-Work XCF line driver supports the commands listed in the following table when the target is the driver. The underlined portion of the command is the minimum abbreviation.

Command	Action
`ALTER` *`driver-parms`*	Dynamically changes the driver configuration. The ALTER command is followed by the driver configuration parameters to be altered. The driver configuration parameters are the same as those specified in the XCF DRIVER Statement. For example: `XCF # ALTER ACCEPTUI=Y` Refer to the specific parameter description for information on possible restrictions about modifying the parameter using the ALTER command.
`HELP`	Display a list of operator commands.
`RESET`	Resets all statistics for the driver. Statistics are printed only if the STATS command precedes the RESET command.
`SHOW`	Displays the current configuration of the driver. The current configuration is always shown automatically following an ALTER command.
`SNAP`	Causes all driver specific control blocks and the driver specific trace table to be snapped (printed in hexadecimal). Link specific control blocks and Entire Net-Work specific control blocks are not snapped.
`STATS`	Causes the immediate printing of statistics and restarts the statistics interval. To print and reset statistics, specify RESET immediately after the STATS command. For example: `XCFD # STATS RESET` Statistics triggered by the STATS operator command are written only to DDPRINT.
`TRACE`	Causes the driver specific trace table to be formatted and printed. The trace table is also formatted and printed automatically when the SNAP command is processed.

Link Commands

The Entire Net-Work XCF line driver supports the commands listed in the following table when the target is a link or all links. The underlined portion of the command is the minimum abbreviation.

Command	Action
`ALTER` *`link-parms`*	Dynamically changes the link configuration. The ALTER command is followed by the link configuration parameters to be altered. The link configuration parameters are the same as those specified on the LINK statement. For example: `XCFD linkname ALTER,PSTAT=Y ↩` Refer to the specific parameter description for information on possible restrictions about modifying the parameter using the ALTER command.
`HELP`	Display a list of available operator commands.
`LOGLON` *`linkname`*	Turns on selective logging for the specified link.
`LOGLOFF` *`linkname`*	Turns off selective logging for the specified link.
`RESET`	Resets all statistics for the link. Statistics are printed only if the STATS command precedes the RESET command.
`SHOW`	Displays the current configuration of the link. The current configuration is always shown automatically following an ALTER command.
`SNAP`	Causes all link specific control blocks and the link specific trace table to be snapped (printed in hexadecimal). Driver specific control blocks and Entire Net-Work specific control blocks are not snapped.
`STATS`	Causes the immediate printing of statistics and restarts the statistics interval. To print and reset statistics, specify RESET immediately after the STATS command. For example: `XCFD linkname STATS RESET` Statistics triggered by the STATS operator command are written only to DDPRINT.
`TRACE`	Causes the link specific trace table to be formatted and printed. The trace table is also formatted and printed automatically when the SNAP command is processed.

Index