CMX/CCP V6.0 (Solaris)

ISDN Communication

Edition June 2003

Comments… Suggestions… Corrections…

The User Documentation Department would like to know your opinion of this manual. Your feedback helps us optimize our documentation to suit your individual needs.

Fax forms for sending us your comments are included in the back of the manual.

There you will also find the addresses of the relevant User Documentation Department.

Certified documentation according DIN EN ISO 9001:2000

To ensure a consistently high quality standard and user-friendliness, this documentation was created to meet the regulations of a quality management system which complies with the requirements of the standard DIN EN ISO 9001:2000.

cognitas. Gesellschaft für Technik-Dokumentation mbH
www.cognitas.de

Copyright and Trademarks

This manual is printed on paper treated with chlorine-free bleach.

Contents

1 Preface

1.1 Brief description of the CCP-ISDN products

CMX/CCP V5.1 (Solaris) provides you with a versatile communications package that supports all the ISDN (Integrated Services Digital Network) services. ISDN communication is implemented using the products CMX, CCP-ISDN-Link and CCP-OSI/NEA (optional) as shown in figure 1. In this manual, the term CCP-ISDN products is used to refer to these products, which are required for ISDN operation.

In addition to using the classic forms of data transfer from your UNIX computer, such as file transfer, you can now also send and receive faxes or navigate through the Internet.

The CCP-ISDN products enable communications to the most varied computer systems from your UNIX computer: other UNIX or TRANSDATA computers, PCs as well as other systems that support the OSI transport services or TCP/IP. In addition, you can communicate with partners in the X.25 network and use the X.25 protocol for end-system-to-end-system connections.

CCP-ISDN products control connection setup, data transfer and connection release using comprehensive security measures. CCP-ISDN requires a suitable Communications Controller (CC) in order to run, and it provides its functions via the Communications Manager in UNIX (CMX).

You can connect with partners in the most varied networks with CCP-ISDN products: connections to NEA, OSI and TCP/IP partners can be set up. CCP-ISDN supports all protocols required for this. The flexible combination of these protocols on one CC allows user-friendly and receiver-compliant operation.

A scalable access control system based on ISDN features protects your computers against unauthorized access from ISDN.

CCP-ISDN supports the protocols DSS1 (Euro-ISDN) and 1TR6 (national German telecommunications protocol).

The UNIX communication products offer an appropriate product structure for the specific applications of communication in the ISDN network.

The following diagram shows the product structure of the various communication options for UNIX computers with ISDN connections:

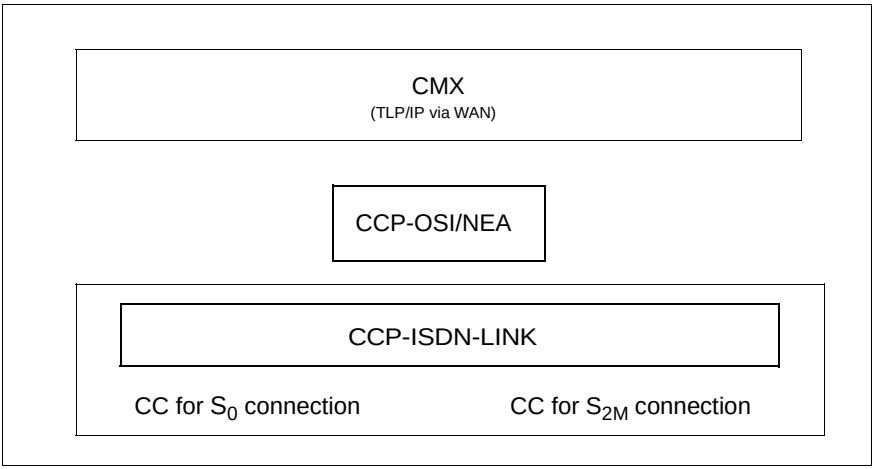


Figure 1: Product structure for ISDN operation

CCP-ISDN-LINK	for operating Communications Controllers for ISDN-S_0 connections (network access software ISDNS0) and ISDN-S_{2M} connections (network access software ISDNS2)
CCP-OSI/NEA	for networking computers with NEA and OSI protocols
CMX	Base product required for operating WAN/ISDN controllers; programming interface for CMX applications; for networking computers with TCP/IP protocols via WAN (SW router) and LAN (RFC1006)

The work involved in configuring the integration of the various links and services is limited. For this reason, the CCP-ISDN products offer a character-oriented user interface (CMXCUI), a component of CMX.

ICMX and XTI are available as program interfaces.

1.2 Target group

This manual is intended for network and system administrators.

A basic knowledge of the Solaris operating system is assumed and familiarity with data communications would be an advantage. You must be familiar with the ISDN options and you should also have a basic knowledge of the CMX functionality.

1.3 Summary of contents

The following section provides you with a summary of the separate chapters in the manual.

The product: The facilities of the CCP-ISDN products are described in chapter “CCP-ISDN - an overview” on page 7. You will be familiarized with the CCP-ISDN access and connection types. The application options are also described: important here are X.25 communications. The protocol selection and addressing options are explained and the interaction with CMX - a requirement for using CCP-ISDN - is also described.

Operation: The chapter “Operating CCP-ISDN” on page 47 explains how to set CCP-ISDN up for operation. It contains instructions for product installation and startup, as well as shutdown and deinstallation. It also contains general instructions on the steps required for configuring CCP-ISDN.

Configuring partner systems: A summary of the FSS parameters relevant to CCP-ISDN is provided in chapter 4, “FSS configuration”, on page 65.

Configuring for different network connections: The chapter “Configuring for different uses” on page 89 provides examples explaining how to configure CCP-ISDN for various networks. It contains a preparation check list and a list of the separate configuration steps in both CCP and CMX. Connections in NEA, OSI and TCP/IP networks are also dealt with, as well as to X.25 subnetworks.

Administration and diagnosis: The sixth chapter lists the commands for administration and diagnosis, the help functions and the commands for the Transport Service Provider (TSP), each arranged in alphabetical order. It also contains information on trace preparation and diagnosis files.

References: A glossary, list of abbreviations, index and list of related publications can be found at the end of this manual.

1.4 Notational conventions

The commands are structured as follows:

– command description
– syntax
– syntax description
– example

The above components are explained in the following section:

Command description

The function of the command is described in the first section of each command description. This section also includes information on the environment in which the command can be used (e.g. entries in files, access authorizations).

Syntax

cmd[␣**-a**][␣**-b**][␣**-c**][␣**-d**␣arg1][␣**-f**␣arg2]␣file␣...

You must enter *cmd* and one or more files for *file*, each separated by a blank. You can additionally input:

– one or more options *-a, -b, -c*. You can input these options separately (**-a**␣**-b**␣**-c**) or together (**-abc**).
– the option *-d*, whereby *arg1* must be replaced by an argument.
– the option *-f*, whereby *arg2* must be replaced by an argument.

The metasyntax used has the following meaning:

Bold characters
: Constants. Characters printed in bold must be input exactly as printed.

Normal characters
: Variables. These characters represent other characters that you select and input.

[ ]
: Options. Arguments in square brackets are optional and can be omitted. Do not input the brackets themselves, unless this is expressly indicated.

␣
: Mandatory blank.

...
: The previous expression can be repeated. If blanks, which are not part of the expression, are required between the repetitions, ... is preceded by a ␣ (blank).

|
: Indicates that you can make a choice. Select only one of the expressions it separates.

underscored
: Default setting

Syntax description

Contains the descriptions of options and arguments (input files, parameters, variables, etc.) that you can input when calling commands. No distinction is made in the running text between constants and variables. All syntax elements, file and path names and commands are shown there in *italics*.

Example

Examples are used to explain the main command function, use of significant options and appropriate combinations of options and arguments. Inputs to the system are shown as bold fixed-width text. All these input lines are concluded with the ↵ key. This is therefore not shown at the end of the lines.

System outputs are shown in *italics* in the running text and as `typewritten text` in all other places.

Notes

This symbol indicates particularly important information to which you must pay close attention.

1.5 Readme and man files

Information on any functional changes and additions to the current product version can be found in product-specific release notices. You will find these notices in the readme package which is supplied with the relevant product.

There are also online manual pages for the CMX/CCP products which can be accessed once the product has been installed.

2 CCP-ISDN - an overview

This chapter explains the basics of CCP-ISDN products and includes a summary of the operational options and functions of CCP-ISDN. The abbreviated name CCP-ISDN will be used in the following text.

CCP-ISDN (Communication Control Program) enables a Solaris system to access an ISDN network. Your Solaris system can communicate with other Solaris or UNIX systems, or with TRANSDATA or SNA systems over an ISDN network. You can also communicate with PCs and other systems, provided they support the ISO transport protocol or TCP/IP. In addition, you can communicate with partners in the X.25 network and use the X.25 protocol for end-system-to-end-system connections.

CCP-ISDN provides you with a powerful transport system for secure data interchange between systems whose applications communicate with each other over the ISDN subnetwork. You can specify the communications protocol to be used for each connection.

A scalable access control system based on ISDN features protects your computers against unauthorized access from ISDN.

CCP-ISDN supports up to two S_0 basic connections or one S_{2M} connection per Communications Controller (see section “Hardware requirements: Communications Controller for ISDN” on page 9) and provides the services of digital data transfer via ISDN.

Several Communications Controllers can be connected in one processor.

You can communicate over the public ISDN network or via a private branch exchange with ISDN capabilities.

CCP-ISDN supports either the DSS1 protocol (Euro-ISDN) or the 1TR6 protocol (German national telecommunications protocol) for signaling. Please ask your network provider which of the two protocols is supported.

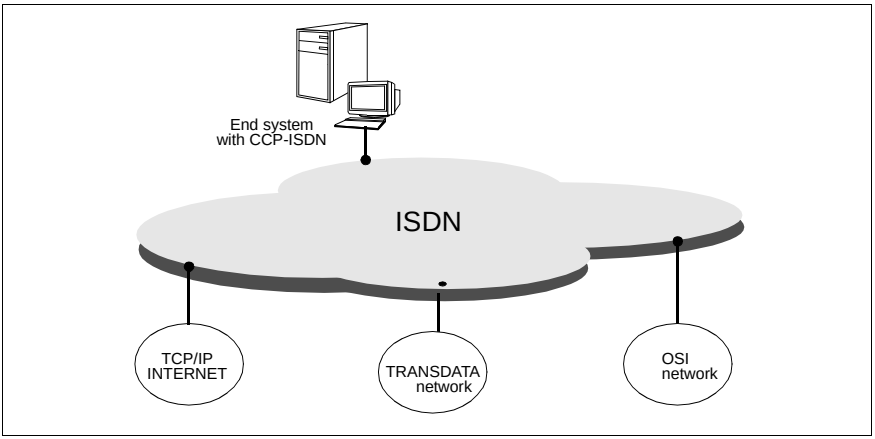


Figure 2: Connections over the ISDN subnetwork

With CCP-ISDN, you can simultaneously operate applications using different protocols with your Solaris system. The following applications can provide system-independent services in the ISDN network using CCP-ISDN:

– Applications that use TCP/IP (in TCP/IP networks, e.g. in the Internet). Together with the product CMX/CCP V5.1, CCP-ISDN supports the transfer of TCP/IP packets using the point-to-point protocol (PPP).
– X.25 applications that use one of the following protocols
  – the ISO transport protocol IS 8073 Cl.0/2
  – the NEA transport protocol
  – TCP/IP
  – or applications that interface directly with the X.25-3 protocol and do not require a transport protocol.
– Applications that use the ISO-transport protocol IS 8073 Cl.0/2 via T.70 (in OSI networks)
– Applications that use the NEA transport protocol (in TRANSDATA networks)

2.1 Software requirements

The base product CMX and the product CCP-ISDN-LINK must always be installed in order to use CCP-ISDN. Additional software products are required, dependent on use. Please see the Release Notice for information on hardware and software version dependencies.

Communications Manager UNIX (CMX)

CMX arbitrates between the current network and applications and provides the network administrator with unified functions for OA&M (operation, administration and maintenance) of CCP and CC.

CMX provides communications services for using applications and communications services in the network. CMX unifies the services of different networks and thereby allows the same application to be used independent of the underlying network.

Both CMX and CCP-ISDN have a character-oriented user interface CMXCUI (CUI=Character User Interface). You can configure and operate CCP-ISDN via menus using CMXCUI.

You do not require any additional products for TCP/IP communication via ISDN. Among other things, CMX allows you to transfer the TCP/IP packets using the point-to-point-protocol (PPP), thus enabling ISDN connections to mobile PCs and to routers such as those from CISCO, 3COM, CONWARE, etc.

2.2 Hardware requirements: Communications Controller for ISDN

Suitable Communications Controller (CC) boards must be installed in the Solaris system to enable communication over ISDN. Each Communications Controller is fitted with plugs for connecting to an S_0 or S_{2M} connection. Each plug on the CC is marked with an access number and must be *physically* connected to an S_0 or S_{2M} connection of its own. The ISDN D- and B- access channels can be used for setting up subnet connections.

The ISDN CCs are loadable CCs, i.e. are controlled by the Communication Control Program (CCP-ISDN-LINK) in order to achieve a *logical* connection to the ISDN network. If two ISDN connections are used, they can be configured independently of one another. It is only possible to set up a connection via a CC after a configuration file is loaded.

The following ISDN connections are supported on a CC:

– CC with $2S_0$ connections

– CC with 1 S_{2M} connection

Please see the Release Notice for information on the Communications Controller type and version, as well as hardware and software version dependencies.

2.3 The ISDN connection

ISDN (Integrated Services Digital Network) covers all communications and information services, such as data transfer and telephony, with just one ISDN connection. There are two types of ISDN connection:

– the S_0 connection with 2 B-channels (switched connections and permanent connections) and

– the S_{2M} primary multiplex connection (also referred to as the S_{2M} connection) with 30 B-channels (each 64 kbit/s) and 1 D-channel (64 kbit/s).

B-channel

B-channels are used for the transparent transfer of user data and provide the digital connection between two communicating partners. The transparent data transfer with ISDN allows the transfer of any format, without the transferred user data being converted or changed by branch exchanges.

The data transfer via an ISDN connection is full duplex, i.e. data can flow in both directions at the same time and with the same priority as on a single physical connection: each partner can use this physical connection for sending and receiving data.

D-channel

With switched connections (see section “Switched connections” on page 11), the D-channel is used as a signaling channel for controlling the setup and release of connections on the B-channels.

The signaling protocol used for transferring connection setup and release control information is a facility of ISDN connections and can be obtained from the network provider. CCP-ISDN supports ISDN connections that use signaling protocols 1TR6 or DSS1:

– DSS1 is the protocol used throughout Europe.

– 1TR6 is the older, Deutsche Telekom standard for German national ISDN and will only be supported until the end of 2005.

An ISDN S_{2M} connection supports only Euro-ISDN (DSS1).

The D-channel can be used for data transfer over permanent connections, as these do not require signaling (see section “Permanent connections” on page 13).

The D-channel can also be used for transmitting packet data over S_0 switched connections.

2.4 Connection types under ISDN

ISDN offers two different types of connection:

Switched connection with

– a switched line to any ISDN partner (SC)

Permanent connection with

– a dedicated line between two defined ISDN partners (PC)

A mixture of switched and permanent lines from the same ISDN connection is not available.

2.4.1 Switched connections

A switched connection can be used to communicate with any ISDN partner that also has a switched connection. Such a connection always consists of two 30 B-channels for switched connections and one D-channel for signaling. You can reach a separate partner on each B-channel simultaneously with a switched connection.

Connections are set up via the D-channel of a switched connection by specifying the subscriber number of the required partner. A B-channel is supplied for this connection as required, by the network provider. Data is transferred over the B-channel. In the case of an S_0 connection, in addition to signaling the D-channel can also be used for transmitting packet data if this has been requested from the network provider.

During pauses in communication the physical connection can be interrupted, and no charges are made during this time (short hold mode). This function must be supported by the control software and the card in the client.

2.4.1.1 Switched connection configurations (bus or point-to-point)

An S_0 switched connection can be configured as a bus or a point-to-point connection by the network provider.

With a bus configuration, several different or same terminal device types can be connected to one ISDN connection. Two devices (one per B-channel) can be used simultaneously.

Only one terminal system can be connected to a point-to-point, e.g. one extension system.

An S_{2M} connection is only available with a point-to-point connection.

2.4.1.2 Combining B-channels with switched connections (multilink)

Two B-channels of an S_0 switched connection can be combined using multilink. The multilink group consists of two B-channels to the same partner. Thus the data throughput can be increased to 128 kbit/s.

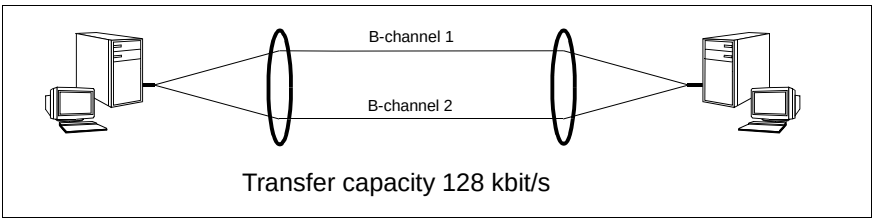


Figure 3: Combining B-channels with switched connections (multilink)

A group of switched connections is set up if a throughput of more than 64 kbit/s is required at connect and both B-channels of the desired partner connection are free. B-channel grouping is requested by entering a correspondingly high throughput (128 kbit/s) at the configuration of the partner connection (in FSS). No further configuration measures are necessary.

The ISO protocol IS7478 is used for sequence control of the data and the individual lines of the group. It must be ensured that the partner system of the multilinking connection also supports this protocol.

If two channels are not available at connect (incoming and outgoing) the request for a connection is rejected.

2.4.1.3 Line costs of switched connections

Particularly in the TCP/IP world, a lot of transport applications have monitoring features, which cyclically poll established connections or partner systems or cyclically distribute information (e.g. routing information). Monitoring takes place semi-automatically without the user's noticing; sometimes appropriate configuration work is necessary.

In LAN segments you can neglect these data transfer, because of the minimal volume. In case of data packet transfer networks like X.25 you can also neglect these short messages because of volume dependent fees.

But if single PCs or LAN segments are connected via WAN (i.e. ISDN) the polling causes line costs. In case of X.25 some network providers charge SVC setup fees. Mainly in case of connection oriented networks (ISDN, dial-up network) the fee is charged according to the frequency and length of the connection. This causes a lot of costs if applications are disadvantageously configured.

In many cases you can switch off or reduce unintentional polling. It is also possible to minimize the number of network connections being maintained without terminating the application if no data is being transferred.

For information on TCP/IP applications see the manual "CMX, TCP/IP via WAN/ISDN" [3].

The line may be busy at connection setup if a switched connection is in use. The B-channel is only assigned to the switched connection for as long as this is maintained. It is then available for other switched connections.

2.4.2 Permanent connections

Permanent connections are connections that are available to the user at all times; there are no connection setup times.

The permanent connection to a defined partner must be requested from the network provider, who then sets up the connection to the desired partner. The transfer protocols must be agreed with the partner when the CCP is set up. Please ask the network administrator of the partner system which protocols (layers 1-4) must be configured for data transfer.

The connection configuration is always point-to-point with permanent connections.

The D-channel is not used for signaling, because a permanent connection does need not to be established. It can instead be used for transferring user data.

Permanent connections are only supported by S_0 connections.

Permanent connections from the network provider allow you to request different types of S_0 access. The Deutsche Telekom offers three different types of S_0 access with permanent connections, using different transfer rates:

Deutsche Telekom designation	**S_0 access with permanent connections**	**Transfer rates**
Digital 64S	1 B-channel	64 kbit/s
Digital S01	1 B-channel and 1 D-channel for user data	64 kbit/s + 16 kbit/s
Digital S02	2 B-channels and 1 D-channel for user data	2 x 64 kbit/s + 16 kbit/s

Table 1: Supported variants of Deutsche Telekom permanent connections (April 1998)

All channels of S_0 access with permanent connections (max. two B-channels and one D-channel) lead to the same partner. This means that only one partner can be reached per S_0 access with permanent connections.

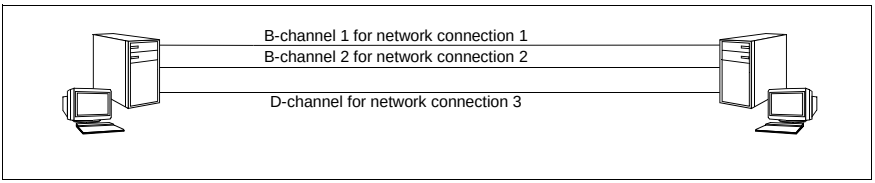


Figure 4: Permanent connections using two B-channels and one D-channel for different network connections to the partner

Several permanent connections to one and the same partner are only meaningful using OSI or TCP/IP. It is only possible to independently set up and use more than one network connection to a single partner with these applications. With the NEA transport protocol, only one network connection can be used at any one time.

2.4.3 Communication options of ISDN connections

The following table shows the options available for the various ISDN connections.

	S_0 connection with 1TR6	**S_0 connection with DSS1**	**S_{2M} connection with DSS1**
Number of channels	2 B-channels 1 D-channel	2 B-channels 1 D-channel	30 B-channels 1 D-channel
Connection configuration	Point to point Bus	Point to point Bus	Point to point
ISDN-subscriber number	a single subscriber number with terminal selection digit	up to 10 multiple subscriber numbers (MSN)	a single subscriber number
Type of connection	PC SC	PC SC	- SC
X.25 via ISDN-PC	PC: PVC + SVC	PC: PVC + SVC	-
X.25 min.	SC: SVC	SC: SVC	SC: SVC
X.25 max. in B-channel	-	SC: SVC	SC: SVC
X.25 max.in D-channel	-	SC: PVC+SVC	-
RA syn	possible	possible	-
RA asyn	possible	possible	possible

Table 2: Communication options for S_0 and S_{2M} connections

2.5 Access control

CCP-ISDN supports a scalable system of access control for ISDN switched connections or X.25 switched connections (SVC) via ISDN permanent connections:

1. “Closed User Group” feature of the network provider
2. Subnet-specific access control
3. Directory number-specific access checks

User-specific access control can also be implemented (point-to-point protocol (PPP) with PAP (Password Authentication Protocol) and CHAP (Challenge Handshake Authentication Protocol)).

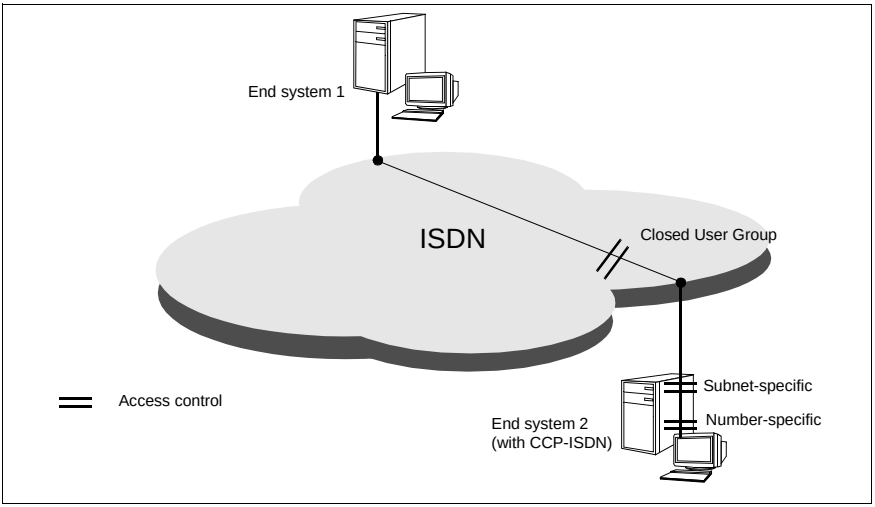


Figure 5: Access control

Closed User Group

CCP-ISDN provides access control based on features of the ISDN and X.25 networks. The “Closed User Group” feature must be requested from the network provider and is subject to charges. Its characteristics and functionality depend on the network operator. Callers who are not members of the closed user group are blocked by the corresponding exchanges in the network. See section “Closed user group” on page 37 and section “Closed and bilateral closed user classes” on page 23.

Subnet-specific access control

Access control can be applied to individual subnet connections or to virtual subnet connections representing a group of individual subnet connections. By explicitly deactivating access control, you grant access to all partners. When you activate access control, you have two options: firstly, a block for all incoming calls from the relevant subnet connection, or secondly, a check of the calling ISDN number or X.25-DTE address for authorization. For the latter case, directory number-specific access control must be configured.

Directory number-specific access control

The main condition for directory number-specific access control is that access control must be active on the subnet connection level. A check is run on the basis of the sender’s directory number, which is sent during connection setup, to verify whether the partner is known (configured directory number). If the partner is known, a further check verifies which type of connection is permitted for this partner (e.g. incoming or outgoing). In any case, only partners for whom incoming connections are authorized are allowed access.

Attempts at access from unauthorized partners are logged together with certain information, such as the date, time and, if known, the sender address. See the description of the DEBUGFILE file in section “Diagnostic files” on page 160.

Subnet-specific and directory number-specific access control can be activated and deactivated without having to modify existing configuration files for CCs, reset subnet connections or reboot the system.

i Please note that this type of access protection has no effect on existing connections during operation.

2.6 Basics of X.25

CCP-ISDN supports communication according to X.25. To explain this, the basic features of X.25 are described here.

The CCITT Recommendations for X.25 describe the access protocol for a data packet transfer network (e.g. the Deutsche Telekom Datex-P network). They describe the lower three layers of the OSI reference model.

The CCITT Recommendations for X.25 are published every 4 years, 1980 (Yellow Book), 1984 (Red Book) and 1988 (Blue Book) (see “CCITT Yellow/Blue/Red Book” [5]). CCP-ISDN supports X.25 communication according to CCITT 1980, 1984 and 1988.

The ISDN network is a line-switched network, i.e a physical line connection is set up for communication between the sending and receiving systems. Continuous user data can only be transferred after the line connection has been set up.

X.25 is a packet-switched network, i.e. single packets of user data are transferred independently to a receiver via virtual connections, i.e. via different routes. The receiving system ensures the completeness, order and integrity of the data.

The connecting component between the ISDN and X.25 networks is called an Access Unit (AU) and is located at the network providers. It provides the Packet Handler Function (PHF) and ensures the correct transition from the line-switched ISDN communication to the packet-switched X.25 communication.

Packet-switched transfers must be configured with their own parameters. These ensure, e.g. correct data stream packet assembly at the interconnection from ISDN into the X.25 network and viceversa, for the correct reassembly of the separate packets into a continuous data stream.

The communication partners of an X.25 connection are the Data Terminal Equipment (DTE) and Data Circuit-Terminating Equipment (DCE) of the X.25 network.

Data Circuit-Terminating Equipment (DCE)

The DCE forms the interconnection from the X.25 network to the DTE.

The DCE is the X.25 protocol partner of the DTE. This means that the X.25 protocol controls the data transfer between the DTE and DCE. The way that the data is passed from the DCE to the network and forwarded in the network, is meaningless to the DTE.

The X.25 variant selected in the menu determines whether your system carries out the control functions of a DCE.

Data Terminal Equipment (DTE)

Both packet and character-oriented DTEs can be connected to an X.25 network.

– Packet-oriented DTEs: can be connected directly to an X.25 network. The X.31 standard defines how a packet-oriented DTE can dial into an X.25 network via an ISDN network (minimum and maximum integration).

– Character-oriented DTEs: are start-stop DTEs (asynchronous stations) that interrupt the data stream during output and transfer, after each character or group of characters. An example of a character-oriented data station is the PC. Character-oriented DTEs require a PAD (Packet Assembler/Disassembler) to connect them to an X.25 network.

X.3, X.28 and X.29 are standards for connecting character-oriented DTEs to an X.25 network. X.25 applications over ISDN, which are to run on a character-oriented DTE, use the X.29 protocol or another, proprietary protocol above layer 3.

Asymmetry between DCE and DTE

The X.25 protocol between DTE and DCE is not symmetric, i.e. the protocol elements passed from the DTE to the DCE have a slightly different structure from those passed from the DCE to the DTE. The DTE and DCE processes differ slightly. For this reason, in the case of ISDN connections without an X.25 network it must be defined which terminal system is the DTE and which is the DCE.

ISO Standards for X.25

ISO Standards IS 7776 (layer 2 protocol description, HDLC-LAPB) and IS 8208 (layer 3 protocol description, X.25) describe the protocols of layers 2 and 3 for data packet transfer.

ISO Standard IS 8208 (X.25) contains an extension with respect to the layer 3 X.25 protocol, that describes the DTE-DTE connection over a line-switched network.

The ISO Standards for the HDLC-LAPB and X.25 protocols comply with the CCITT Recommendations for X.25. They differ in their approach to the X.25 recommendations. The data packet transfer is described with respect to the network viewpoint in X.25 and from the DTE viewpoint in the ISO Standard.

2.6.1 Virtual connections

The data between the sending and receiving DTE is transferred over a virtual connection. This differs from a physical connection in that the virtual connection is not exclusively assigned a physical line (the DTE connection to the network), but rather only for the transfer duration of each separate packet. This allows a physical line to be used simultaneously by a number of virtual connections.

The DTE X.25 interconnection to the Access Unit can be assigned a number of logical channels, so-called channel ranges.

There are two types of X.25 virtual connections:

– Switched Virtual Call (SVC). Is set up and released as required with special control packets. It only occupies a logical channel for the time the virtual connection exists.

– Permanent Virtual Circuit (PVC). Two DTEs are permanently assigned to each other. The PVC is agreed with the network provider. The connection is always present on a fixed logical channel and is therefore not set up or released.

The ISDN connection routes to the Access Unit are assigned X.25 integration-specific facilities when the X.25 connections are configured. The number of logical channels and their distribution in the channel ranges are examples of X.25-specific facilities. Freely selectable X.25 facilities, which normally apply for all connections through this point, can additionally be assigned to the X.25 interconnection during configuration. Information about freely selectable X.25 facilities can be found in section “X.25 facilities” on page 22.

Logical channel

Logical channels are used for addressing virtual connections. Each virtual connection is assigned just one logical channel. Each channel has a channel number.

– Channel numbers 1 to 4095 are available for virtual connections.

– Channel number 0 is reserved for system messages (not in the French “TRANSPAC” X.25 network).

– The number of channels must be agreed with the network provider when connection is made to an X.25 network. Up to 255 logical channels can be assigned to a main connection in the Deutsche Telekom Datex-P network. Channel numbers above 255 are also used outside Germany.

Several logical channels can be assigned to the X.25 transfer of the DTE to the Access Unit, so-called channel ranges.

The available logical channels are split into one range for the PVC and one for the SVC.

SVC channel range: this range can itself be split into logical channel ranges:

– only incoming connections
– incoming and outgoing connections
– only outgoing connections

Channel range order: the channel range numbers must be assigned in the following order:

Channel range: permanent < incoming < incoming/outgoing < outgoing

The channel ranges could, for example, be requested from the network provider as follows:

– two channels for permanent connections (PVC) [1-2],
– three channels for incoming connections (SVC) [3-5],
– one channel for incoming/outgoing connections ((SVC)[6-6]
– and four channels for outgoing connections (SVC) [7-10].

2.6.2 X.25 facilities

The following facilities can be assigned during configuration to an ISDN permanent connection with X.25 or to the switched connection from a specific ISDN connection to an Access Unit. They must be agreed with the partner.

Window size

The window size defines the number of unacknowledged packets (i.e. the number of packets that must be followed by an acknowledgment) that can be sent to the remote DTE or received from the remote DTE. The window size can be defined differently for each transfer direction.

The default window size is 2.

Packet length

The packet length defines the maximum permissible length of the user data field per data packet.

The default packet length is 128 bytes.

This facility allows the source data packet length for a main connection to be defined which deviates from the default length. The source packet length defines the maximum permissible length of the user data field per data packet for the main connection. The source packet length can be defined differently for each transfer direction. A packet length of between 16 and 1024 bytes can be defined (in power 2 steps).

Acceptance of reverse charging

If this facility is agreed, incoming calls with reverse charging are allowed.

Extended packet consecutive number

This facility allows the sent packets consecutive number P(S) and the received packets consecutive number P(R) to be extended from modulo 8 (default) to modulo 128.

The numbering is 0 to 7 with modulo 8 and 0 to 127 with modulo 128. With the extended packet consecutive number, seven bits are used for numbering the packets instead of the default 3 bits.

Timer and retry counter

With CCP-ISDN, you can define a number of timers and retry counters for X.25 communications. This is only necessary in exceptional cases and requires a comprehensive knowledge of X.25.

Closed and bilateral closed user classes

In the case of closed user classes only the members of this class can communicate with one another, i.e. a DTE which belongs to a user class can only set up connections to a DTE which belongs to the same user class. A DTE can be a member of several user classes. The closed user class facility enables a closed user network to be created within the X.25 network for reasons of data security.

– A bilateral closed user class is a closed user class to which only two DTE belong.

– A class with outgoing access allows connection requirements to be sent to a DTE outside the timesharing class.

– A class with incoming access permits an incoming call from a DTE outside the class to be answered.

Fast Select

The Fast Select facility allows individual packets to be sent and received. Individual packets are packets with which 128 byte instead of 16 byte long user data can be transported.

Individual packets are suitable for the exchange of small data quantities. This facility exchanges the data in the packets at connect and disconnect.

The “Fast Select” facility must be agreed with the network provider by the participating DTEs.

The called DTE can react in two ways:

- The remote DTE can answer the incoming call with an extended packet call acceptance (length of field for user data = 128 bytes). Then the called DTE can clear the connection with a clear request or use the connected call to exchange data packets. If the DTE clears a connection, it can use the clear request packet to transfer another 128 byte user data.
- The remote DTE answers the incoming call with an extended packet clear request (with up to 128 bytes user data).

Agreement of the throughput class

The outgoing throughput class is specified for all selected virtual connections. The outgoing throughput class can be changed for each individual selected virtual connection in the connect phase if the DTE has agreed the agreement of the throughput class facility.

Agreement of the flow control operand (window size, packet length)

Flow control operands determine the size of the sending or receiving window of the DTE and the largest permitted packet length which can also be selected separately for both transfer directions. If the agreement of the flow control operands facility is configured for a DTE, it can negotiate and select the flow control operands for the relevant selected virtual connection in the connect phase.

Configuring the facilities

Facilities can be specified for both an ISDN connection with X.25 integration and for the DTE address of a particular X.25 partner.

The facilities in the configuration file are specified for the ISDN connection. In Forwarding Support Service (FSS) the partner-specific facilities are specified.

2.7 Communication with X.25 partners via an ISDN switched connection

CCP-ISDN supports three different options for X.25 communication

– connection to an ISDN partner with X.25 communication in the B-channel (DTE-DTE link)

– connection to partners in a public or private X.25 network with the minimum integration according to X.31 case A.

– connection to partners in a public or private X.25 network with the maximum integration according to X.31 case B (Packet Mode Bearer Service).

2.7.1 X.25 communication via DTE-DTE connection

The partners are connected via ISDN and communicate in the B-channel in accordance with the X.25 protocol. This allows applications that were conceived for an X.25 network to be used in the lower-cost ISDN network as well.

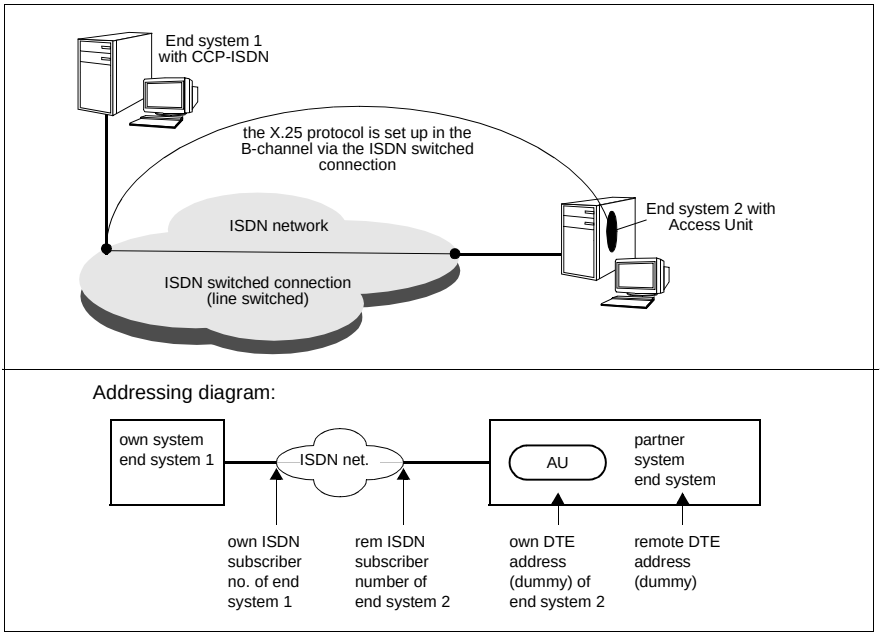


Figure 6: X.25 communication with partners in ISDN networks: DTE-DTE connection

The X.25 network is not used with an X.25 connection to an ISDN partner. This is then a pure DTE-DTE connection in which the Access Unit and partner, end system 2 in the example, are congruous, so to speak.

The two DTE addresses are then merely dummy addresses and can be agreed freely between the two DTEs. The remote ISDN subscriber number of end system 2 represents the subscriber number of an X.25 data terminal device and consists of digits. The Access Unit is not used because the X.25 partner has a direct ISDN connection. The DTE-DTE connection type is therefore always adequate because there is no X.25 network between the partners.

For DTE-DTE connections you can specify how many X.25 switched connections (SVCs) are set up for each B-channel. See section “User-defined number of X.25 switched connections (SVC) via a B-channel” on page 121.

2.7.2 X.25 minimum integration

X.25 minimum integration in accordance with X.31 case A makes it possible to reach a partner system which only has a X.25 connection via an ISDN connection. This type of X.25 data packet connection via ISDN is called two-step dialing.

– In the first step, an ISDN connection is set up from your system to an Access Unit (AU).

– Then in the second step, as soon as the ISDN connection exists, one or more X.25 connection are created transparently by the AU for the ISDN network.

The AU, which is maintained by the network provider, serves as an instance that enables a transfer from the ISDN network to packet transfer network. This can be a packet transfer with one ISDN entry and one X.25 exit.

In the case of X.25 minimum integration only SVCs and no PVCs are permitted.

The channel number ranges are assigned by the X.25 network provider (see also section “Basics of X.25” on page 18).

In the case of communication with partners in the X.25 network, ask your network provider for the ISDN subscriber number of the Access Unit.

For the minimum configuration you can specify how many X.25 switched connections (SVCs) are set up for each B-channel. See section “User-defined number of X.25 switched connections (SVC) via a B-channel” on page 121.

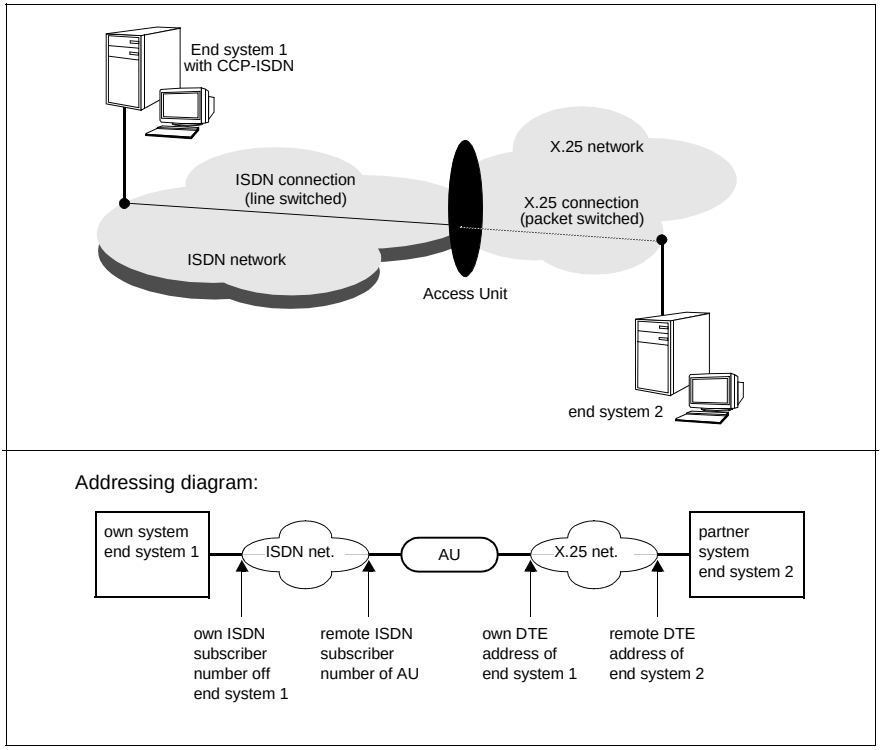


Figure 7: Communication with partners in the X.25 network: two-step dialing

Configuration for unknown partners in the case of X.25 minimum integration

With this configuration option you can specify defined access to your system for unknown partners, i.e. partner systems not managed in your system. This applies to incoming calls.

Configurations for unknown partners are effective only when access control is inactive, since unknown partners will not otherwise have access to the system.

2.7.3 X.25 maximum integration

With X.25 maximum integration according to X.31 case B, the ISDN network provides the user with packet-oriented services according to X.25 on request. Unlike X.25 minimum integration (see section “X.25 minimum integration” on page 27), where the connect is carried out in two steps, maximum integration permits the direct transfer of X.25 data via the ISDN connection.

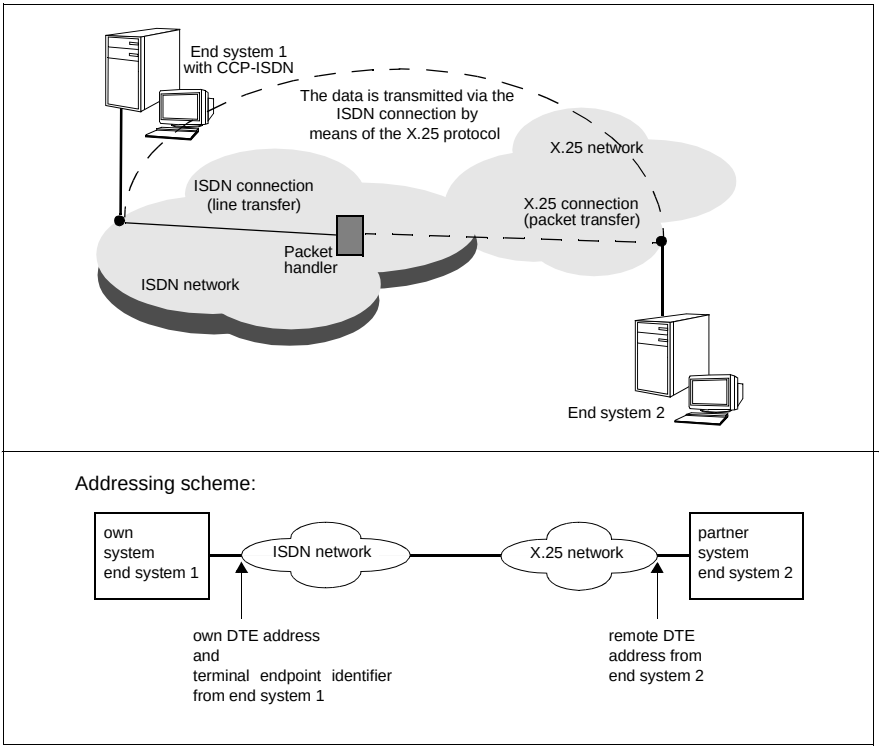


Figure 8: Communication with partners directly in the X.25 network

This functionality must be requested from the network provider of the ISDN connection. It can be provided either in the D-channel or in the B-channel.

If you request X.25 access via the D-channel, you receive a “X.25 user in the D-channel” ID from the ISDN switching center. This ID is called a ”Terminal Endpoint Identifier”. The transmission capacity for X.25 data in the D-channel is 9.6 kbit/s; the rest of the transmission capacity of the D-channel is reserved for ISDN signaling.

In the case of X.25 access via a B-channel a switched connection to a packet handler is first created if necessary. Here the ISDN switching center is simply informed at setup that X.25 functions are needed. After assigning the ISDN connection it can be used like an exiting X.25 main connection. The transmission capacity is 64 kbit/s.

In the case of a S_{2M} connection X.25 access is only available via the B-channel for the maximum integration.

The properties of X.25 access must be agreed with the network provider and defined during the configuration of the ISDN connection.

2.8 Point-to-point protocol for TCP/IP routing

CCP-ISDN supports the point-to-point protocol (PPP).The point-to-point protocol is described in RFC 1171 and RFC 1172 and is a standardized method with which datagrams (i.e. data packets over TCP/IP that contain sender and receiver addresses) can be encapsulated and routed.

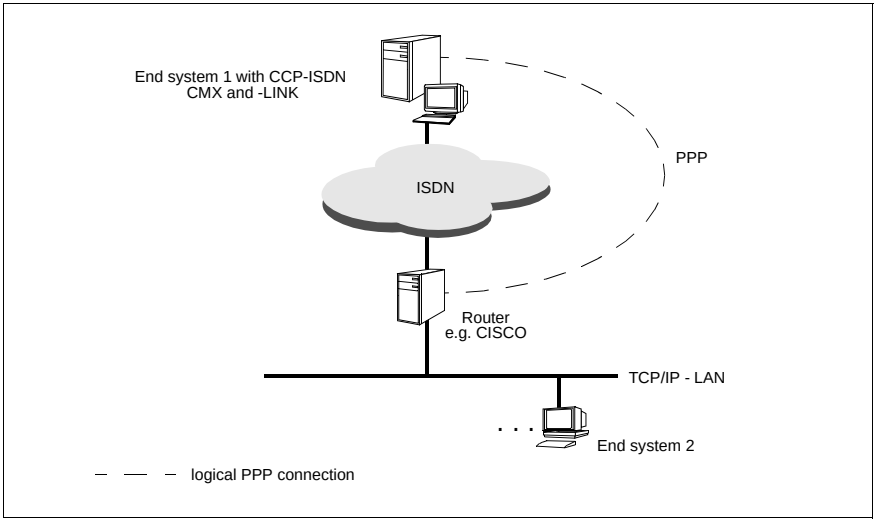


Figure 9: LAN/WAN routing with CMX uses PPP

The point-to-point protocol (PPP) is required by many routers for communication with other routers. If you use your Solaris system together with the CMX/CCP software as a router, PPP enables communication to routers from CISCO, 3COM, CONWARE and others.

When you use Mobile Computing you need asynchronous PPP. This allows the connection of GSM PCs to an RM Server over ISDN. You get a complete digital connection to notebooks in the GSM network.

Further information can be found in the manual “CMX, TCP/IP via WAN/ISDN” [3].

2.9 Subnet connections

CCP-ISDN provides the functions “Selection of an alternative subnet interface” and “Multiport”.

With the selection of an alternative subnet interface, connections can alternatively be set up via different subnet interfaces with different directory numbers. This gives your subnet failure security.

The multiport function groups different subnet interfaces together in a single, virtual ISDN port (multiport) under a single directory number. Multiport provides more than two B-channels under one directory number and for outgoing connections uses the same directory number for all the ports of the multiport.

2.9.1 Selection of an alternative subnet interface

With the function “Selection of an alternative network connection” you can optimize the connectivity of your system with its partner systems in the ISDN network. If a line is engaged or a there is a failure, the desired connection is rerouted to an alternative connection.

An alternative network connection can be used in the case of local failures and failure of the partner system.

Engaged or failed local connection

If a connection cannot be set up due to an engaged line or problems with a local connection, CCP-ISDN tries to set up the connection via an alternative local connection. A requirement for this is that you have defined several ISDN switched connections with the same subnet ID in the CC configuration. The individual connections can (but must not necessarily) be distributed on different controllers.

Selection of an alternative subnet interface can also be used for WAN connections, such as dedicated lines (see the manual “CMX/CCP, WAN Communication” [4]).

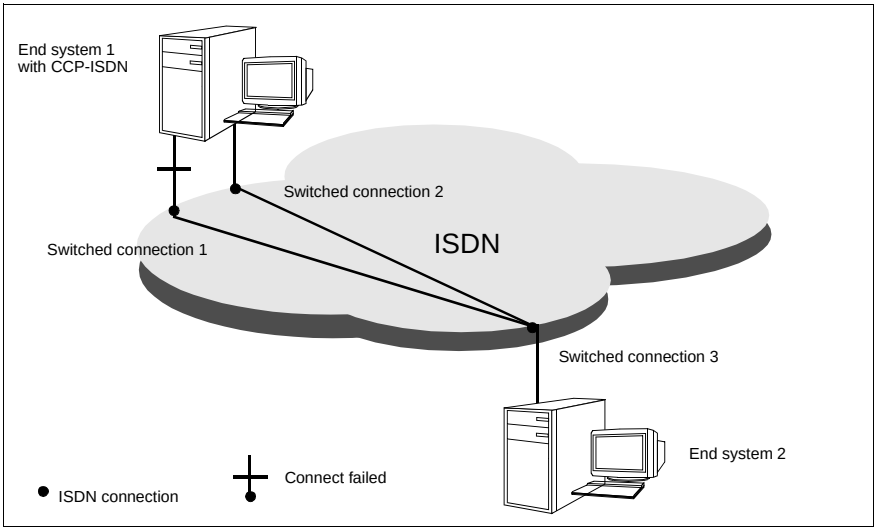


Figure 10: Using an alternative network connection in the case of problems with the local connection

Engaged or failed partner connection

This function is not valid for the OSI transport profile ISDN-CONS.

If your partner's connection is engaged or there is a failure and your partner has another ISDN connection, CCP-ISDN attempts to reach the partner via this connection. To do this you must have defined several routes in FSS for the partner connection (SNPAROUTES objects).

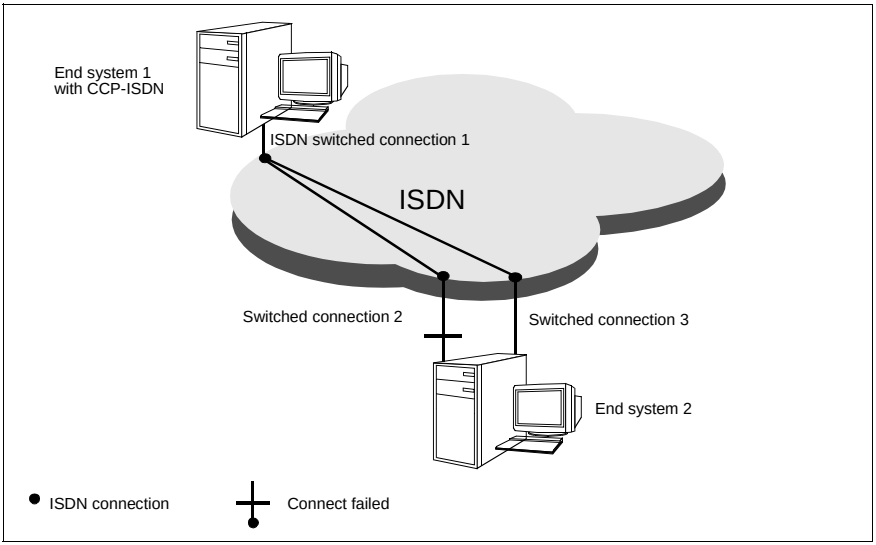


Figure 11: Using an alternative network connection in the event of problems with the remote connection

2.9.2 Multiport

Access to an ISDN network from a computer can be set up via several physical ports. Physical ports can be grouped together as a single virtual port (multiport) under a single directory number. Each port then has the same port identification, consisting of a directory number and a subnet ID.

For outgoing calls the same sender address is always used. For the partner systems this means that they must manage only one entry for the multiport system. For incoming calls more than two B-channels are available to the multiport, thus enhancing the accessibility of the system considerably.

This option of building multiports is based on a network provider facility. At Deutsche Telekom, this is called “One directory number for several PMxAs (primary multiplex ports) or/and BaAs (basic access lines)”. This feature must be requested from the network provider. It can be set up for point-to-point ports.

The individual ports making up an ISDN multiport are handled in the same way for administration and routing purposes. Changes to the configuration of a multiport, i.e. the addition or removal of a local port, do not affect the global and local network configuration in the FSS and TNS in the local and remote computers. They relate only to the controller configuration file.

2.10 NEA routing

NEA routing is the capability of forwarding the incoming NEA data packets which are not intended for the own subnetwork. A system that can do this is called an Intermediate System (IS).

In a NEA computer network you can operate your system as end system (ES) and at the same time as intermediate system. In this way a dedicated system is no longer necessary for routing.

The NEA routing can be switched on and off using the menu and command interface (see also section “Switching NEA routing on and off” on page 158). The default setting is OFF, i.e. NEA routing is not activated. Your system then functions as end system.

Processing of incoming data packets

Incoming data packets are processed independently of their NEA target address as follows:

- If the target address is the address of the own NEA system, the packet of the local application (if registered) is delivered.
- If the target address is the address of another NEA system, the packet is forwarded.
- If the target address is not known, the packet is discarded. The target address is unknown if
  - no entry exists in the Forwarding Support Service (either in the form of a NSAP or a GNSAP object) or
  - the application is not requested or active.

2.11 Euro-ISDN (DSS1) and German national ISDN (1TR6)

For S_0 connections CCP-ISDN supports both the signaling protocol DSS1, which is also used in Euro-ISDN, and the German national ISDN 1TR6. For S_{2M} connections only DSS1 is supported. You must ask your network provider whether 1TR6 or DSS1 is used on your ISDN connection.

– DSS1: the public network providers in Europe have all agreed to DSS1. This ensures trouble-free data interchange within Europe. New connections are always set up with DSS1, which also offers the option of signaling the OSI reference model layers one to four via the D-channel. CCP-ISDN uses this option. The term “Euro-ISDN” is used below to refer to ISDN signaling with DSS1.

– German national ISDN: the German national ISDN using 1TR6, standardized by Deutsche Telekom, is a predecessor of Euro-ISDN and will only be supported by Deutsche Telekom until 2005. The most important difference to Euro-ISDN is that the protocols of the OSI reference model layers one to four cannot be signaled over the D-channel.

2.11.1 Closed user group

CCP-ISDN automatically supports the function “closed user group in the ISDN network” for both Euro-ISDN and German national ISDN, if the network provider has set this function up for you.

The network providers offer the option with ISDN of setting up a closed user group (CUG) for each service such as teletex, telephone or data transfer. This makes it possible to build up a closed user network within the ISDN network, for data security reasons, because only members of a closed user group have access to the group subscriber numbers.

Only the members of a closed user group can communicate with each other for data transfer, for example, i.e. a system that belongs to a specific closed user group can only set up connections to a system that belongs to the same closed user group. One system can be a member of several closed user groups.

You must apply for membership to a closed user group with your network provider. The network provider registers the addresses of the closed user group members and enables the functionality.

2.11.2 ISDN subscriber number

Each ISDN connection has one or more ISDN subscriber numbers. These numbers are assigned by the network provider.

All ISDN subscriber numbers are made up of digits, no other characters are allowed. You input your own ISDN subscriber number without the local exchange code for the CC configuration. The remote ISDN subscriber number of the partner must *always* include the local exchange code, even with local calls, and must additionally include the country code for international calls.

Example for the remote subscriber number of a partner system in Munich, Germany:
08912345
Example for the remote subscriber number of a partner system in Vienna, Austria:
0043112345

Subscriber identification is a special feature of the ISDN network. With each connection setup the ISDN shows the subscriber being called the subscriber number of the person calling, guarantees its correctness and therefore makes efficient access protection possible.

The difference between the Euro-ISDN and German national ISDN subscriber numbers are explained in the following section.

Euro-ISDN: Multiple Subscriber Number (MSN)

You can obtain up to ten subscriber numbers for each Euro-ISDN S_0 connection from the network provider. These subscriber numbers are known as multiple subscriber numbers (MSN). All MSNs must be entered in the CCP-ISDN configuration file because the sender address is mandatory in the ISDN network for subscriber identification purposes.

An S_{2M} connection is only ever assigned one subscriber number.

German national ISDN: ISDN subscriber number with terminal selection digit

With 1TR6, you receive one ISDN subscriber number from the network provider for each ISDN connection. You then have to add a suffix digit of your choice (0...9) to this number. This digit is called the terminal selection digit and is used to identify your terminal device if more than one (similar) device is connected to your ISDN connection.

The ISDN subscriber number must be entered together with a terminal selection digit in the CCP-ISDN configuration file.

2.11.3 Signaling options

One special feature of the ISDN network is the transparent data transfer in the B-channel which does not define a specific format for the user data, i.e. it does not specify a user data protocol. It is of no consequence to the ISDN network whether the data is, e.g. telematic data according to protocol T.70 or packet data according to X.25. The terminal devices themselves must ensure that the data is correctly interpreted.

The protocols required by a communication relationship are dependent on both the communication relationship itself (e.g. applications with or without X.25), and the network architecture of the systems that are communicating with each other (e.g. NEA architecture in the TRANSDATA network, TCP/IP protocols in the Internet architecture).

With switched connections, it is possible to set the required protocols via the ISDN signaling. If this is not possible, the protocols must be configured. This is always necessary with permanent connections.

Signaling options with switched connections

The protocols are assigned connection-specific for each new switched connection. The protocols are assigned to the B-channel when the connection is set up. The protocol signaling options differ between Euro-ISDN and German national ISDN.

See section “CCP profile” on page 44 for information on the possible protocol combinations on layers 1 to 4 of the OSI reference model.

Euro-ISDN: Signaling options with switched connections between systems using CCP-ISDN and Euro-ISDN

Euro-ISDN supports signaling of the transfer and communication protocols from layers 1 to 4 with switched connections. When the connection is set up, CCP-ISDN signals:

– whether it is data with or without X.25

– whether another proprietary protocol is used for matching character-oriented terminal devices to the packet-oriented X.25 network

– whether it is the transport protocol of an OSI network, NEA network or TCP/IP.

CCP-ISDN also uses this protocol signaling option with incoming switched Euro-ISDN connections, resulting in automatic setting of the protocol information when the connection is set up between systems that both use CCP-ISDN.

Euro-ISDN: signaling options with switched connections to partner systems not using CCP-ISDN

Partner systems not using CCP-ISDN do not always use the protocol signaling option for setting the user data protocols.

The protocols can be specified partner-specific for partners in the FSS.

German national ISDN: signaling options with switched connections

The protocols cannot be signalled if you have a German national ISDN connection (1TR6 signaling protocol). The transport and user data protocols must always be configured in the FSS.

Summary of the signaling options

The following table summarizes the signaling options for different cases:

	Euro-ISDN to partners with CCP-ISDN	**Euro-ISDN to other partners**	**German national ISDN with 1TR6**
Switched connections	Protocols are signalled (no protocol configuration necessary)	Protocols must be configured if the signaling is not successful	Protocols must always be configured and apply for the complete S_0 connection
Permanent connections	Protocols must always be configured		

Table 3: Summary of the signaling options

Signaling Light

With the Signaling Light function, protocol information is suppressed in signaling for a Euro-ISDN port during active connection setup. This is necessary for partner systems which cannot evaluate this information correctly and therefore reject the call. These systems can thus accept the connection request correctly. See object class FACIL, attribute *isdn-partner-prot=SIMPLE* on page 69.

2.12 Architecture of CCP-ISDN

The communication software for data transfer in ISDN consists of the network access software and various Transport Service Providers (TSPs). The TSP and network access software together form the transport system for ISDN data communication. The network access software together with a specific TSP are also called CCP profile.

The network access software supports the linking of your system into a public ISDN network or to an ISDN private branch exchange (e.g. Hicom). It executes the communication tasks independently on CCs with one or two ISDN connections up to and including layer 3a of the OSI reference model (subnet access layer).

A TSP implements the transport protocol in layer 4 and a protocol of the subnet-independent part of the network layer (layer 3c).

CMX makes the transport applications independent of the transport system used.

From the viewpoint of the Solaris system, access to a CCP profile is via the ICMX, TLI, Sockets and XTI interfaces.

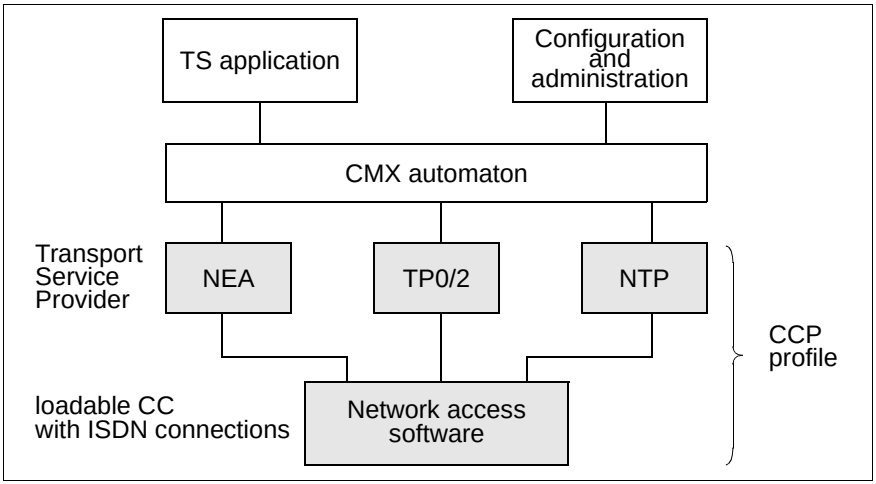


Figure 12: TSPs and the network access software

Network access software ISDNS0 and ISDNS2

The network access software controls the CCs for the ISDN subnetwork. The configuration file (CF) for the network access software defines the characteristics of your local ISDN connection: e.g. your own ISDN subscriber number, the protocols to be set during connection setup and the X.25 characteristics of the interconnection into an X.25 network or to an X.25 partner in ISDN.

The network access software assigned to the CC is loaded onto the CC together with the assigned configuration file when the system is started up.

Transport Service Provider

A Transport Service Provider (TSP) provides the network access software with a specific transport protocol and the relevant transport services for the different network architectures. The TSPs are configured in the kernel and are controlled by the CMX automaton (see the “CMX, Operation and Administration” User Guide [1]). As an administrator, you can define the use of a specific TSP for the communication of applications.

– NTP for X.25 communication:
  NTP (null transport) is set as TSP for CMX applications that offer direct access via ISDN to the network service of the X.25 subnetwork. Thus an X.25 application must run directly on the X.25 layer 3 protocol.

– TP0/2 for OSI architecture (connection-oriented):
  TP0/2 is the TSP for an OSI environment with OSI transport service and OSI protocols. The TP0/2 uses its information on the protocols and partner subscriber numbers in the Transport Name Service and Forwarding Support Service.

– NEA for the TRANSDATA-NEA architecture:
  The NEA-TSP provides the NEA transport service in the TRANSDATA network. The NEA-TSP uses its information on the NEA protocols and partner subscribers in the Transport Name Service (see section “Transport Name Service (TNS) and Forwarding Support Service (FSS)” on page 46).

2.12.1 CCP profile

A CCP profile is designated as being an ordered number of protocols that are arranged logically above each other in the four layers of the transport system, thereby defining specific network characteristics. The communication options are defined by the combination of protocols. See the “CMX, Operation and Administration” User Guide [1] for the basics of CCP profiles.

The protocols define behavioral rules and formats between the partners. Two instances that belong to the same layer thus communicate using a common protocol. This means that all systems that support the protocols listed below for the transport system are possible partners for your Solaris system.

The CCP-ISDN products support five CCP profiles that are implemented via the ICMX interface. ISDN-NEA, ISDN-NX25, ISDN-X25, ISDN-CONS (Connection-Oriented Network Service) and ISDN-HDLC.

The following table illustrates the possible CCP profiles.

Product	**CCP profile**	**Type of connection**
CCP-ISDN-LINK	ISDN-X25	X.25 applications via ISDN
CCP-ISDN-LINK	ISDN-X25	TCP/IP applications with X.25 protocol via ISDN
	ISDN-HDLC	TCP/IP applications with or without T.70 protocol via ISDN
CCP-ISDN-LINK CCP-OSI/NEA	ISDN-NEA	NEA applications via ISDN
	ISDN-NX25	X.25 applications with NEA protocols via ISDN
	ISDN-CONS	OSI applications with T.70 protocol or with X.25 protocol

Table 4: Overview of the CCP profiles

2.12.2 Relationship between the CCP profiles and TNS address formats

The address formats of your application are administered in the CMX Transport Name Service (TNS). The address formats depend on the TSPs and therefore also on the CCP profile to be used for communication. This is irrelevant to you if you administer the applications via the CMX menu, as the menu internally assigns the address formats to the profile.

The following table shows the address formats for the CCP profiles that are required if you edit the TNS entries for the local and remote applications in a file.

ISDN-NEA	**ISDN-NX25**	**ISDN-CONS**	**ISDN-X25**
Address format= WANNEA		Address format= WANSBKA	Address format= WAN3SBKA

Table 5: CCP profiles and the corresponding address formats of the local and remote applications

2.12.3 Protocol selection

CCP-ISDN attempts to set the correct protocol combination for each connection, using the signaling information. In those cases where the signaling information is insufficient, you can make entries in the Forwarding Support Service (FSS) for the relevant partner.

The following priorities therefore apply to protocol selection:

1. Protocol signaling has top priority, i.e. if protocols are signaled when a switched connection is set up, CCP-ISDN always uses these signaled protocols.
2. If the signaling is insufficient, the information from the FSS is evaluated.

If the protocols are not configured with one of the above options, the default TCP/IP with HDLC is used.

2.13 Transport Name Service (TNS) and Forwarding Support Service (FSS)

In addition to configuring the network access software, you must supply the TNS and FSS with the necessary address information in order to operate your CCP profile and set up communication connections to remote systems.

Transport Name Service (TNS)

The TNS administers information about the TS applications in the network.

Each network and each transport system requires special address information in order to be able to address the communication partner. CMX provides the Transport Name Service (TNS) with which you can administer the names and addresses of TS applications.

The TNS administers the address information in the TS directory. It is stored there for the local and remote TS applications.

Forwarding Support Service (FSS)

The FSS administers information about computers in the network.

The FSS supplies the transport service providers with information for setting up network connections. The FSS describes how the remote system can be reached from the local system: over which subnetwork, with which DTE address, etc.

The system can obtain this information from the TNS in many cases. If the TNS information is not sufficient or there are no TNS entries (e.g. with TCP/IP connections), the missing information is supplied by the FSS.

3 Operating CCP-ISDN

You must complete the following steps in order to work with CCP-ISDN:

- Startup of the Communications Controllers

  1. Initially describe your local ISDN connection. A menu interface is available for this. CCP-ISDN stores the configuration data in a configuration file (CF).
  2. Assign the configuration file and the network access software ISDNS0 or ISDNS2 to a previously selected Communications Controller (CC).
  3. Finally load the configuration file and the network access software onto the Communications Controller.

  The ISDN connection is ready for operation after you have completed these steps. Information on the individual steps can be found in section “Setting up the ISDN access” on page 49.

- Configuration of partner systems

  Then you must additionally obtain parameters for the partner systems and use these to configure the corresponding partner systems. Summarized information about these steps can be found in section “Configuring partner systems” on page 55.

  You can find examples in this manual in chapter “Configuring for different uses” on page 89.

  A detailed description of configuring partner systems can be found in the “CMX, Operation and Administration” User Guide [1].

- Entry of transport system applications

  Once you have configured the partner systems, you must enter your transport system applications in the Transport Name Service (TNS) (see section 3.7, ”Entering transport system applications”, on page 61).

This chapter contains instructions for installation, shutdown and deinstallation, in addition to information for operating CCP-ISDN.

3.1 Installation

The CMX/CCP products comprise one or more packages in accordance with UNIX SVR4. If you have expert knowledge of the interdependencies between the packages you can install the packages separately.

CCP profile	**Products to be installed**	**TSP**
ISDN-NEA	CMX CCP-ISDN-LINK CCP OSI/NEA	NEA
ISDN-NX25	CMX CCP-ISDN-LINK CCP OSI/NEA	NEA
ISDN-CONS	CMX CCP-ISDN-LINK CCP OSI/NEA	TP0/2
ISDN-X25	CMX CCP-ISDN-LINK	NTP

Table 6: Overview of the products to be installed

Under Solaris, the CMX/CCP software products can be installed, updated and deinstalled during normal system operation, i.e. there is no need to reboot the system.

Installation of the CMX/CCP products always includes startup of the installed components. If the software is being updated or deinstalled, the components involved have to be shut down first. In some cases, you will have to terminate all CMX applications beforehand and, if appropriate, start them again after installation has been completed successfully.

CCP-ISDN is installed using the Webstart wizard. The installation CD is automatically mounted when you place it in the drive.

- ► Click on one of the README icons to read the product-specific Release Notices before you start installation.
- ► Click on the *Installer* icon.

  A welcome window is displayed.

- Click on *Next*.

  A series of windows guides you through the installation procedure.

- The Installation Summary window is displayed at the end of the installation procedure. Click on Details in this window for information on whether or not installation was successful, diagnostic information, as well as information on how to proceed if an error occurs.

When the installation procedure has been completed successfully, the CMX components are integrated in the operating system kernel.

The functions of CMX/CCP can be used immediately.

Please refer to the Release Notice for more information on hardware and software dependencies and installation.

3.2 Setting up the ISDN access

You must configure your local ISDN connection before you are able to start it up. Both the initial startup and restarting are meant in this case. Initial startup takes place immediately after installation of CCP-ISDN and restarting after each ISDN configuration change.

CMX must be in operation for the configuration with CMXCUI.

You must carry out the following steps to set up your local ISDN connection:

1. After installing CCP-ISDN, configure your ISDN connection by creating a configuration file (CF). The configuration file describes your configuration. You can find information on this topic in section “Loading network access software and configuration file” on page 54.
2. Assign your configuration file and the network access software to the previously selected Communications Controller (CC). This step must be carried out for initial startup and after each configuration change. Information on this can be found in section “Assigning network access software and configuration file to a CC” on page 52 and in the “CMX, Operation and Administration” User Guide [1].
3. Load the network access software onto the selected CC; the assigned configuration is also loaded automatically. This step must also be carried out for both initial startup and after each configuration change. Relevant information can be found in section “Loading network access software and configuration file” on page 54.

3.3 Creating a configuration

You must configure the network access before you can use CCP-ISDN. You describe the local ISDN connection characteristics and select CCP-ISDN characteristics with the configuration of the subnet profile. CCP-ISDN saves the configured parameters in a configuration file (CF).

The assigned configuration file is also loaded when ISDN is started on a Communications Controller. The configuration file then determines the operating characteristics of your ISDN connection.

You can create a configuration file in two ways:

- Configuring with the menu system (see section “Editing configuration files with the menu system” on page 51).

  Menu-controlled configuration has the same result as a standard configuration. The menu system functions cover all the normal application cases for the subnetwork configuration.

  You can find examples for creating a configuration file with the menu system in chapter “Configuring for different uses” on page 89.

- Configuring in expert mode by editing a configuration file with the configuration-oriented generator language KOGS. This method should only be used in special cases and requires special knowledge. You can find a detailed description of KOGS in the Release Notice for CCP-ISDN-LINK.

Note:

We recommend that you proceed as follows when creating a configuration file that has special cases to take into account:

– Initially create a configuration file with the menu system.

– Then modify the file to your requirements by editing the existing KOGS macros.

3.4 Editing configuration files with the menu system

The menu system for editing configuration files provides you with an FMLI-oriented user interface and is part of the CMX menu system (CMXCUI).

With the menu system, you can carry out all the activities required for editing a configuration file. You can, for example, create new configuration files, modify or delete existing ones or create backups.

General instructions on the menu system (CMXCUI) can be found in the “CMX, Operation and Administration” User Guide [1].

3.5 Assigning and loading the configuration file

After the configuration file has been created and successfully compiled, you should

- assign the network access software and the configuration file to a CC
- load the network access software and the configuration file.

3.5.1 Assigning network access software and configuration file to a CC

After selecting a Communications Controller, you must assign the network access software and a configuration file to it.

If more than one CC is being operated, you must select the network access software ISDNS0 or ISDNS2 and corresponding configuration file for each CC and then assign them to their respective CCs.

Once the configuration file and the network access software have been assigned, they are automatically loaded and started with each system startup, thus avoiding the necessity of an explicit start.

You can use either the menu system options (CMXCUI) or administration commands to assign the network access software and the configuration file.

Assigning with the menu system

Proceed as follows when working with the menu system:

1. Open the main menu of the CMX menu system.
2. Select the *CCs - Communications Controller*...menu option.
3. The *State of Communications Controllers* menu appears with information on the corresponding CC (name and status of the CC) and information on the network access software and configurations which may have already been assigned and loaded. Select the desired Communications Controller (CC).

Then the *Operations for CC* menu is displayed. You can select various menu options, e.g. you can load or unload the selected CC or change a configuration or go into expert mode.

– If you wish to assign network access software and/or a configuration file, select the menu item *Change Configuration*. Enter ISDNS0 or ISDNS2 and the name of the configuration file assigned to the CC in the following menu.

– If the CC already has other network access software or another configuration file assigned to it, this network access software/configuration file must first be unloaded. Select the *Unload*... menu option for this and only subsequently assign the new network access software/configuration file.

Assigning with administration commands

You can also make the assignment with administration commands. A detailed description of using administration commands can be found in chapter “Administration and diagnosis commands for CC” on page 127. The following two commands are also described in detail there. Proceed as follows:

- After selecting the Communications Controller, the *Operations for CC* menu is opened. Select the *Enter expert mode* menu option.
- You assign the network access software to a CC with the *Assign* command. This assignment is only effective after loading.
- You assign a configuration file to a CC with the *Exchange* command.

3.5.2 Loading network access software and configuration file

CCP-ISDN is normally loaded when the operating system is started. In addition, other options are available.

Loading with the menu system

You can load CCP-ISDN during system operation using the menu option *Load CC...* in the CMX menu *Operations for CC*.

Loading with administration commands in expert mode

You can load network access software with the corresponding configuration file (CF) on a CC using the *Load* command. A detailed description of this can be found in chapter “Administration and diagnosis commands for CC” on page 127.

If network access software is loaded on the CC, any previously loaded software on the same CC is stopped. Existing links are broken. If you are working in the menu system a message will be displayed which will require you to either confirm or cancel the action.

3.6 Configuring partner systems

After configuring the local ISDN connection, you must enter information for each of the partner systems you want to reach with the ISDN connection: this includes a definition of the partners in the Transport Name Service (TNS) and Forwarding Support Service (FSS), in addition to the configuration of the local ISDN connection in the configuration file.

You can, for example, provide information about the remote subnetwork interface and define the route you want to use to reach the partner. These entries are administered by the CMX component FSS. Information about the local and remote applications as well as network addresses are administered in the TNS.

A detailed description of the configuration of partner systems can be found in the “CMX, Operation and Administration” User Guide [1]. The following description of the individual steps required for configuring partner systems is only intended to give you a brief overview.

You can carry out the FSS and TNS entries via the CMX menu options. For special cases or mass data entry, as the expert you can also enter data directly.

To configure individual partner systems with CMXCUI, carry out the following steps:

1. Define the route to the remote subnet interfaces. Information on the process using the menu system can be found in section “Defining routes to remote subnet interfaces via the menu system” on page 56.
2. Then enter information about the partner system. Information on the process using the menu system can be found in section “Entering information about the partner system via the menu system” on page 57.
3. Finally input the transport system application (see section “Entering transport system applications” on page 61).

3.6.1 Defining routes to remote subnet interfaces via the menu system

First define a route to your respective partner system (remote subnet interface). You define a route by its end points, in other words the local subnet ID and the SNPA address of the remote computer.

If a remote computer has more than one SNPA address, there is more than one route to it in the same subnet.

Proceed as follows:

1. Select the *SNPAROUTES - Routes to Remote Subnetwork Interfaces...* menu option from the CMX main menu.
2. All existing routes are then displayed.
   - If the route you require is present, select it.
   - If the route you require is not present, press the ENTER key to open the *Operations on routes* menu. Select the *Create* option. Input the routing information required for reaching the partner system. This includes the SNPA address of the partner system and the subnet ID, which defines the local SNPAs via which the partner system can be reached (see the “CMX, Operation and Administration” User Guide [1]).
     - Input a symbolic name for your route.
     - Input the type of subnet address of the remote subnet interface from the viewpoint of the local system. A list of all possible subnet types is displayed if you press the CHOICES key.
     - Input the subnet ID (SNID) that describes the local subnet interface. A list of all possible values is displayed if you press the CHOICES key. The subnet ID that you select here must be defined in your configuration file.
     - Enter the subnet address of the remote subnet interface.
     - Specify whether you want to enter facilities. Then enter your facilities in the subsequent windows.

3.6.2 Entering information about the partner system via the menu system

You must now input some parameters for your partner system. Proceed as follows:

- Select the *NSAPs* menu option in the CMX main menu. All NSAPs that are present are displayed.
  - The configuration is finished if the NSAP you require is present.
  - Proceed as follows if you want to modify an existing entry for a remote NSAP:
    - Use the arrow keys to select the name of an NSAP that you want to edit. Press MARK and then ENTER. This opens a menu from which you can select *Change*.
  - Press ENTER if the required NSAP is not present. This opens the *Operations on Remote Systems* menu. Select the *Create* option. Input the partner system network addresses (see also the “CMX, Operation and Administration” User Guide [1]). Note the close relationship between transport system applications and remote NSAPs. The name of an NSAP object must be input as the transport address for some transport system applications. Proceed as follows to configure a network address:
    - Input a name for the partner system (up to 32 printable characters). You will need this name later when entering the applications.
    - Input the type of the network over which the partner system can be reached from the local system viewpoint (NEA, Internet, OSI).
    - Input the NSAP address.
    - Input the method by which the subnetwork address (SNPA address), which is required for routing to the partner system, is to be determined:
      - *via intermediate system* - the partner system is reached via an intermediate system for which determination of the SNPA address is defined.
      - *via routing protocol* - the SNPA address is determined by means of the ARP routing protocol.
      - *via static route* - the SNPA address is input directly for this partner system.

from NSAP address - The corresponding subnetwork address is encoded in the network address (NSAP address) and shall be taken from there.

3.6.3 Entering access control via the menu system

Proceed as follows:

1. Select the SUBNET menu option *in the CMX main menu*.
2. Press the ENTER key. The *SUBNET objects* menu then opens if objects already exist. If you want to create a new object, do not mark an object, but press ENTER.
3. The *Operations ...* menu then appears. Select the option *Create*.
   - Enter the access control information you require.
     - Specify the subnet ID which also represents the SUBNET object.
     - For the *Allowing incoming calls* option use the CHOICES key to specify whether or not you want to permit incoming calls.

       DEFAULT: access control is not activated. If you are using the CS-Route product then incoming calls may or or may not be permitted depending on the *admit* parameter of the corresponding route.

       NONE: blocks all incoming calls

       ALL: allows all incoming calls

       RESTRICTED: enables you to activate directory number-specific access control
     - The specification of the *DTE name* does not affect the access control.
4. If you want to activate subnet address-dependent access control (you have selected the value *RESTRICTED* for the *Allowing incoming calls* option), you must now specify the type of call that is permitted for the route: incoming calls, incoming and outgoing calls, outgoing calls or no calls. Select the menu option *SNPAROUTES - Routes to Remote Subnetwork Interfaces... from the CMX main menu.*

5. A list of all the existing routes is then displayed.
   - If the route you require already exists, select it from the list.
   - If the route you require does not exist, press the ENTER key. The *Operations on Routes to Remote Subnetwork Interfaces* menu appears. Select the *Create* menu item. Enter the routing information you need to access the partner systems: the name, the type of the remote subnet address, the subnet ID, the remote subnet address (depending on the type of remote subnet address, e.g. ISDN address), facilities= YES. (See the “CMX, Operation and Administration” User Guide [1]).
   - When you have entered all the routing information, press SAVE to save your entries. You will now see the “Facilities” window for the route.
   - In the *Facilities to Route* window there is an *Admission* option, under which you can specify the types of calls to be allowed: incoming calls (*INCOMING_ONLY*), outgoing and incoming calls (*BOTH_IN_AND_OUT*), no calls (*NEITHER_IN_NOR_OUT*), only outgoing calls (*OUTGOING_ONLY*) or the default value (-) No value assigned to the parameter.

3.6.4 Entering the configuration for unknown partners in the case of X.25 minimum integration via the menu system

Proceed as follows:

1. Select the SUBNET menu option *in the CMX main menu*.
2. Press the ENTER key. The *SUBNET objects* menu then opens if objects already exist. Do not mark an object, but press ENTER.
3. The *Operations ...* menu then appears. Select the option *Create*.
   - Specify the subnet ID which also represents the SUBNET object (e.g. ISDN-22).
   - The *Admit incoming calls* option must be filled with *DEFAULT* for the configuration of unknown partners. Access control must not be active.
   - Specify the “DTE name” that you defined in the configuration file under the *DTE Name* option. X.25 minimum integration is now set and the description of the X.25 access selected for incoming unknown calls at subnet interface ISDN-22.

     Note: you should set the name under *DTE Name* for all X.25 connections via ISDN. You must specify it if two or more X.25 network accesses have been configured in the corresponding KOGS by means of the XZSTW-KOGS macro.
     If only one X.25 network access has been configured in the KOGS using XZSTW then you can omit this specification since this X.25 network access is used as the default.

This configuration does not affect incoming calls from known partners, nor outgoing calls.

3.7 Entering transport system applications

You must now enter your transport system application after having configured your local ISDN connection and the respective partner systems.

You should be familiar with the basic concept and terms used before starting entry of the transport system application.

A detailed description can be found in the “CMX, Operation and Administration” User Guide [1].

3.7.1 Entering transport system applications via the menu system

The description in this section only represents a brief overview of the separate steps in the CMX menu system.

Proceed as follows:

1. Select the *TSAs - Transport system application* menu option from the CMX main menu. You administer local and remote TS applications with this option.
2. First input the GLOBAL NAME of the TS application. Entry of all name parts is optional. You can use the CHOICES function to select from existing names.
3. Then carry out the following entries in the *Operations on TS applications* menu:
   - Input the local TS application first by selecting the *Assign local name* menu option.
   - A form is subsequently displayed that contains a field for each transport system type over which your application is to communicate. Assign a *LOCAL NAME* to your required local TS application. Local applications have a *LOCAL NAME* that consists of a number of T-selectors for the various types of transport systems. The meanings of the CCP profiles and corresponding address components are described in the “CMX, Operation and Administration” User Guide [1].
   - Use the *Assign TRANSPORT ADDRESS* menu option to input the remote TS application.

– This opens a menu from which you can select the type of transport system with which the TS application is to be reached. After the transport system has been selected, a form appears in which you must enter either the complete address information or the name of an NSAP object.

This completes the remote partner system and transport application configurations.

3.8 Shutdown

Carry out the following steps to shut CCP-ISDN down:

1. Stop the network access software
   – in the menu system (CMXCUI) using the corresponding function of the *Communications Controller* menu.

   A detailed description of the operation of the menu system can be found in the “CMX, Operation and Administration” User Guide [1].
2. Remove the assignment of the network access software to the CC
   – in the menu system by the entering the value ’-’ in the appropriate input field.

In this way the network access software is no longer automatically loaded at system startup.

3.9 Deinstallation

The configuration files are saved when CCP-ISDN is deinstalled and are employed again when a reinstallation is carried out. The configuration files that were last assigned to a CC are thereby automatically compiled during the reinstallation.

Please refer to the Release Notice for more information on hardware and software dependencies and deinstallation.

4 FSS configuration

The description of an FSS configuration is stored in a database called the Forwarding Support Information Base (FSB). The FSB is an object-oriented database. The FSS defines a series object classes with their attributes. By making an entry in the FSB you create an object which belongs to a particular object class and to which you assign attribute values according to its object class.

For standard configurations you can make the required FSB entries via the character-oriented user interface CMXCUI. Sample configurations can be found in chapter “Configuring for different uses” on page 89.

The *fssadm* command serves to configure the Forwarding Support Service (FSS) in expert mode. The command *fssadm* should only be used for special configurations and only by people with comprehensive knowledge of CMX and ISDN.

FSS and the *fssadm* commands are described in detail in the “CMX Operation and Administration” User Guide [1]. This chapter only deals with the attributes and values of object classes which are relevant for CCP-ISDN.

Description format

The description format of the *fssadm* commands described below for the various object classes can be found in section “Notational conventions” on page 4 of this manual.

Actions

When you use the *fssadm* command, you apply specific actions to the object classes and their attributes. Refer to the description of the object class concerned to determine which actions are permitted for a specific object class.

The following actions can be specified in the *fssadm* command:

create
: Creates an object with the specified attribute values.

delete
: Deletes an object.

set
: Sets the attributes of an object to the specified values.

get

Retrieves the objects of the specified object class together with their attributes.

If attribute values are specified, *fssadm* only selects objects with these attribute values. *fssadm* rejects specification of multiple attributes which each uniquely identify an object.

Examples of fssadm commands

```
fssadm create SNPAROUTES name=RX25_CS1 subnet=X25-1\
              dteaddr=12345
fssadm create NSAP name=NSIP_CS1 internet-addr=205.75.2.10\
               snpa-list=RX25_CS1
```

Refer to the man pages for a detailed description of the *fssadm* command.

Uppercase and lowercase with actions, object classes and attributes

fssadm does not discriminate between uppercase and lowercase or between a dash (-) and underscore (_) when naming actions, object classes, attributes and attribute values which are symbolic constants.

Examples of using uppercase and lowercase with fssadm commands

The following three entries are permitted and have the same value:

```
subnet=ISDN-1
SubNet=isdn-1
SUBNET=ISDN_1
```

Uppercase and lowercase with the name attribute

Uppercase and lowercase are discriminated for names which you have to define yourself (object class FACIL, SNPAROUTES, NSAP with the name attribute).

Examples of using uppercase and lowercase with the name attribute

The following three entries designate three different names:

```
name=HOST-1
name=Host-1
name=host_1
```

Uppercase and lowercase with fssadm information output

In the *fssadm* outputs, the names of actions and attributes are written in lowercase and the names of object classes and attribute values which are symbolic constants are written in uppercase.

Help functions

You can use the following commands to call help on the *fssadm* syntax:

fssadm␣?
: Displays a general description of the *fssadm* syntax and information on the help function.

fssadm␣action␣**?**
: Displays the object class to which an action may be applied.

fssadm␣action␣object class␣[[attribute name=]attribute value ...]␣**?**
: Completes the command with the attributes which are suitable for the specified context. The context is thereby only taken into account for the attributes which follow the context in the output.

fssadm␣action␣object class␣[[attribute name=]attribute value ...]
attribute name=␣**?**
: Outputs the syntax of the specified attribute in the specified context. Only those attributes in the specified context are taken into account which precede the queried attribute.

Example

The `fssadm create snparoutes type=isdn-nc ?` command returns the following output:

```
fssadm create SNPAROUTES <name> [<subnet>] type=ISDN-NC\
              {<remsnpa> <nailed-up-isdn> } (min=0,max=1)\
               [<facil>]
```

The `fssadm create snparoutes subnet=isdn-1 type=?` command returns the following output:

```
<type>: ISDN | ISDN-[NC] | X31-M[SA] | X31-S[VC] | X31-P[VC]
```

The special meaning of the question mark (?) for the shell must be noted when this is input. The character may have to be nullified with a backslash (\).

Abbreviated notation

Abbreviated keywords:

You can abbreviate the keywords as long as the commands, actions, object classes and attributes remain unique.

Example

```
fssadm create snparoutes name=XY subnet=isdn-1 isdn-nr=132345
```

Abbreviated notation:

```
fssadm cr sn nam=XY su=isdn-1 is=132345
```

Abbreviations with setting parameters:

Attribute keywords can be omitted as long as the values can be identified uniquely by their setting, format or context.

Example

```
fssadm create snparoutes name=XY subnet=isdn-1 \
               type=isdn isdn-nr=132345
```

Abbreviated notation:

```
fssadm cr sn XY isdn-1 132345
```

Keyword and setting parameters can be combined as required.

Avoid the attribute and setting parameter notation in shell scripts which are to remain applicable over lengthy periods. Such commands may be rendered invalid if new object classes, attributes or attribute values are introduced in a product update.

4.1 FACIL object class

You can assign facilities to each route using a FACIL object. Parts of these facilities, such as those which are the same for several routes, can be grouped in an additional FACIL object, which you can then assign to the directly assigned FACIL object by means of the *facil* attribute. If a facility is defined both in the additional FACIL object and in the directly assigned object, the facility of the directly assigned FACIL object takes precedence.

4.1.1 ISDN facilities

fssadm␣create␣FACIL␣name=[**␣isdn-cug=**][**␣isdn-ra=**]
[**␣isdn-partner-prot=**][**␣isdn-throughput=**]

name=
Name of the FACIL object

1-15 characters: letters, digits, the characters '_' and '#'.
Upper case and lower case are distinguished. The first character must not be a digit or an underscore '_'.

isdn-cug=
Selection of a closed ISDN user group

Decimal number between 0 and 65535

isdn-ra=
Process for adapting the transmission rate. This is not supported by all CC types and interfaces (see release notice)

X[30/V110-SYN]
Rate adaptation synchronous in accordance with X.30/V110 (not S2)

V[110-ASYN]
For GSM mobile computing

isdn-throughput=
Throughput in kbit/s

9.6 64 128

isdn-partner-prot=

Used to adapt the ISDN signaling

1TR6/TYPE1

The partner has an 1TR6 connection (national ISDN) and uses the service indicator “X.21 service” for signaling to set up a data transfer without the X.25 protocol. Data transfer with the X.25 protocol is signaled with the service indicator “DÜ64 kbit/s”.

1TR6/TYPE1A

The partner has an 1TR6 connection (national ISDN) and uses the service indicator “DÜ64 kbit/s” for signaling to set up a data transfer without the X.25 protocol. Data transfer with the X.25 protocol is signaled with the service indicator “X.21 service”.

SIMPLE

Protocol information is suppressed for signaling. The value *SIMPLE* is required when the partner cannot evaluate the signaling information for the protocols of the various layers correctly when an attempt is being made to set up an incoming call, and the connection cannot therefore be established.

4.1.2 X.25 facilities

fssadm␣create␣FACIL␣name=
[␣**x25-octet-string=**][␣**x25-packet-size=**][␣**x25-window-size=**]
[␣**x25-throughput=**][␣**x25-cug=**][␣**x25-cug-oa=**][␣**x25-bcug=**]
[␣**x25-revch=**][␣**x25-transit-delay=**][␣**x25-fast-select=**]
[␣**x25-rpoa=**][␣**x25-nui=**]

name=
Name of the FACIL object

1-15 characters: letters, digits, the characters '_' and '#'.
Upper case and lower case are distinguished. The first character must not be a digit or an underscore '_'.

x25-octet-string=
DTE facilities according to CCITT X.25 Annex G (ISO8208)

1...109 octets in hex format

x25-packet-size=
Packet size in the send and receive directions in the form:

Send direction/[receive direction]

Possible values for send and receive directions:

16, 32, 64, 128, 256, 512, 1024, 2048

If no value is specified for the receive direction, the value of the send direction is used as a default.

x25-window-size=
Window size. Number of unacknowledged data packets which can be sent to the network or received from the network in the form:

Send direction/[receive direction]

Possible values for send and receive directions:

1...127

If no value is specified for the receive direction, the value of the send direction is used as a default.

x25-throughput=

Transfer speed in the send and receive directions in the form:

Send direction/[receive direction]

Possible values for send and receive directions:

2.4 4.8 9.6 19.2 48 64

If no value is specified for the receive direction, the value of the send direction is used as a default.

x25-cug=

Selection of a closed X.25 user group

0...9999

Leading zeros are evaluated: 1-2 digits indicate “basic format”, 3-4 digits indicate “extended format”.

x25-cug-oa=

Selection of a closed X.25 user group which can also access subscribers outside the group.

0...9999

Leading zeros are evaluated: 1-2 digits indicate “basic format”, 3-4 digits indicate “extended format”.

x25-bcug=

Selection of a bilateral closed X.25 user group

A bilateral closed user group is a user group to which only 2 DTEs belong.

0...9999

Leading zeros are ignored. The “extended format” always applies.

x25-revch=
Reversed charges request or request for charge acceptance

B[OTH_REQ_AND_ACC]
Reversing of charges is requested and request for charge acceptance is accepted.

R[EQUEST_ONLY]
Reversing of charges is requested.

A[CCEPT_ONLY]
Request for charge acceptance is accepted.

N[EITHER_REQ_NOR_ACC]
Neither reversing of charges is requested nor is the request for charge acceptance accepted.

x25-transit-delay=
Desired transfer time in milliseconds

0-65534

x25-fast-select=
Set Fast Select transfer

The *Fast Select* facility allows data up to a maximum length of 128 bytes to be sent and received in the packets for connection management by using an extended Call User Data field.

If this attribute exists, Fast Select transfer is requested in the connection request packet.

N[O_RESTRICTION]
The partner can answer with the call acceptance or release request packet.

R[ESTRICTION]
The partner may answer only with the release request packet.

x25-rpoa=
Selection of a route via one or more transition networks which are identified by your DNIC (Data Network Identification Code) in the form:

DNIC[+DNIC...] with a maximum of 12 elements

x25-nui=

Network User Identification

The Network User Identification (NUI) is an identification issued by the network provider which is entered in the Call Request packet by the DTE during an outgoing X.25 connection setup.

Maximum 16 printable characters (ASCII or EBCDIC) or hexadecimal character pairs in the form:

formid:nui-value

formid

A
: The NUI is coded in ASCII.

E
: The NUI is coded in EBCDIC.

X
: The NUI follows in hexadecimal format.

nui-value
: A character string issued by the network provider

4.1.3 Facilities for X.25 via an ISDN dial-up connection

In addition to the X.25 facilities described in the previous section, the facilities explained below are also relevant.

fssadm␣create␣FACIL␣name=
[␣**x31min-svc-to-Bchan=**][␣**x25-description=**]

name=
Name of the FACIL object

1-15 characters: letters, digits, the characters '_' and '#'.
A distinction is made between upper and lower case. The first character must not be a digit or an underscore '_'.

x31min-svc-to-Bchan=
Seizure by SVC of the B-channels to an ISDN partner; only for X.25 minimum integration or DTE-DTE links.

n-TO-EACH
where n={1..127}

A maximum of *n* SVCs is set up via a B-channel. Once this number has been reached, a new B-channel (if available) is used in any further call request.

MAX-TO-EACH
The SVCs are set up via the one B-channel, up to a maximum number of SVCs (defined in the XZSTW macro). Any additional SVCs are set up via the next B-channel up to the maximum number of SVCs.

MAX-TO-ONLY-ONE
Default case: the SVCs are set up via the single B-channel, up to a maximum number of SVCs (defined in the XZSTW macro). All other SVCs are rejected, since no other B-channel is used.

x25-description=
Refers to the predefined description of an X.25 access (DTE name) in the configuration file.

The value must match the value of the DTE-NAME operand in the configuration file.

Possible values:

1...8 characters: letters, digits, characters '$', '#' and '@'.
The first character must not be a digit. Case is unimportant.

You should set an x25-description for all X.25 connections via ISDN. It must be specified if two or more X.25 accesses are configured in the corresponding KOGS by means of the XZSTW-KOGS macro. If only one X.25 network access is configured in the KOGS via XZSTW then you can omit this specification since this X.25 network access is used as the default.
In the event of an inconsistency, an entry is generated in the debug file of the corresponding controller!

Please note that all SNPAROUTES objects of this kind with the same subnet ID (attribute *subnet*) and the same ISDN address (in the attribute *x31-msa=isdn-nr/...*) must be assigned the same value for *x25-description*.

In X.25 maximum integration the x25-description attribute affects only the outgoing calls. It is required only if the X.25 access description is not assigned under its own DTE address by specifying the optional address section in the corresponding SNPAROUTES object. *fssadm* issues a warning if this rule is violated.

4.1.4 Subnet-independent facilities

fssadm␣create␣FACIL␣name=[␣**admit=**][␣**npid=**][␣**ppp-profile=**]
[␣**t70-profile=**][␣**compress=**][␣**ppp-accm=**]

name=
Name of the FACIL object

1-15 characters: letters, digits, the characters '_' and '#'.
A distinction is made between upper and lower case. The first character must not be a digit or an underscore '_'.

admit=
Admittance of incoming and outgoing calls. This attribute, combined with the SUBNET attribute *incoming-call*, can be used to configure partner-specific access control.

B[OTH_IN_AND_OUT]
Incoming and outgoing calls are admitted.

I[NCOMING_ONLY]
Only incoming calls are admitted.

O[UTGOING_ONLY]
Only outgoing calls are admitted.

N[EITHER_IN_NOR_OUT]
Neither incoming nor outgoing calls are admitted.

npid=
ID of the network protocol/service

I[nternet]
TCP/IP protocols

P[RIVATE]
Other protocols

N[EA]
NEA protocol

OSI-CO[NS]
Connection-oriented OSI network service for TP0/2 profile

ppp-profile=

Use of point-to-point protocol

N[O]

The point-to-point protocol is not used.

S[TANDARD]

The point-to-point protocol is used.

G[SM]

The asynchronous point-to-point protocol is used. Used with mobile computing. The rate adaptation “V.110 asynchronous” with a transmission speed of 9.6 kbit/s is applied. The following FACIL attributes are set automatically:

isdn-throughput=9,6

isdn-ra=V110-ASYN

t70-profile=

Use of the T.70 protocol variant

Y[ES]

The T.70 protocol variant is used.

N[O]

The T.70 protocol variant is not used.

compress=

Specifies whether Van Jacobsen header compression is to be performed

T[CP/IP]

The TCP/IP header is compressed.

N[O]

Compression is not performed.

ppp-accm=

Use of asynchronous control character mapping. With this attribute you define which control characters will be transmitted transparently for asynchronous PPP.

Only when ppp-profile=GSM

“value[+value+...]”

“value” is a control character

See also the “CMX, TCP/IP via WAN/ISDN” manual [3].

4.2 SNPAROUTES object class

You use an SNPAROUTES object to configure a route within a subnet. The route is defined by its starting point and end point. The starting point of the route is a local subnet connection, the end point is the subnet connection of the remote system. Several connections can be grouped together locally if they lead to the same subnet. The starting point of the route is then defined by a subnet ID, which groups together the required subnet connections.

The attributes and attribute values described below are meaningful for ISDN switched or permanent connections.

fssadm␣create␣SNPAROUTES␣name=␣subnet=[␣**type=**]␣address
[␣**facil=**]

name=

Name of the SNPAROUTES object

1-15 characters: letters, digits, the characters '_' and '#'.
A distinction is made between upper and lower case. The first character must not be a digit or an underscore '_'.

subnet=

Subnet ID of the subnet connections belonging to this route (or group of routes). The value corresponds to the parameter value for SUBNID in the KOGS macro XSNID.

ISDN-i

i=1...32
For subnet access via ISDN switched connections (with or without the X.25 protocol) or via ISDN permanent connections (without the X.25 protocol)

X25-i

i=1...32
For subnet access via ISDN permanent connection with the X.25 protocol

type=

SNPA-address type
This attribute must be specified only if address information is missing (e.g. for type=ISDN-NC). Otherwise, it is determined implicitly from the address specification. This attribute is displayed for output and can be specified as a filter criterion in the get command. The following values are possible:

ISDN
ISDN switched connection

ISDN-N[C]
ISDN permanent connection

X31-M[SA]
X.25 connection according to X.25 minimum integration or DTE-DTE link

X31-S[VC]
Selected virtual X.25 connection (SVC) according to X.25 maximum integration in the B-channel or D-channel

X31-P[VC]
Permanent virtual X.25 connection (PVC) according to X.25 maximum integration in the D-channel

X25
X.25 connection (SVC) via ISDN permanent connection

PVC
X.25 connection (PVC) via ISDN permanent connection

address

Address of the remote subnet connection. One (and only one) of the following parameters must be specified except for permanent connections (type=ISDN-NC).

isdn-nr=
Remote ISDN number of the partner including the local access code

Maximum 20 digits

nailed-up-isdn=
CC and line number of an ISDN permanent connection in the form:

CC number[/line number]

CC number
Number of the CC: 1...32

Line number
Line number: 00, 01, 02, 32, 33, 34

This attribute is optional. For the selection of a local line in an outgoing connection request, only the *subnet* attribute is decisive, not the CC number/line number.

x31-msa=
Two-step dialing address for a connection according to X.25 minimum integration or DTE-DTE link in the form:

isdn-number/x25-dte-address

isdn-number
ISDN number of the X.25-DCE (1st step) or of the X.25-DCE including local access code
Maximum 20 digits

x25-dte-addr
X.25-DTE address (2nd step)
Maximum 17 digits

dte-addr=
Remote X.25-DTE address (for type=X25, therefore X.25 protocol via ISDN permanent connection)
Maximum 17 digits

pvc-nr=
(For *type=PVC*, therefore X.25 protocol via ISDN permanent connection) X.25-PVC number and optionally the related local DTE address in the form:

PVC number[/DTE address]

Possible value for PVC number: 0...4095

Possible value for DTE address: 1-17 decimal digits

x31-dte-addr=
DTE address of the partner for a switched connection according to X.25 maximum integration and optionally the DTE address of own connection in the form:

x25-dte-address-rem[/x25-dte-address-local]

Per DTE address maximum 17 digits

x31-pvc-nr=

Defines the address of a partner which is reached via a permanent virtual X.25 connection (PVC) in X.25 maximum integration. Such a connection is only possible in the D-channel. The address is specified in the following form:

number[/x25-dte-address-local]

number

PVC number of the partner
0...4095

x25-dte-address-local

Own DTE address
Maximum 17 digits

For the selection of a local line in an outgoing connection request, only the *subnet* attribute is decisive, not the optional X25-dte-address-local.

facil=

Refers to an object in the FACIL object class

Name of a defined FACIL object

1-15 characters: letters, digits, the characters '_' and '#'.
Upper case and lower case are distinguished. The first character must not be a digit or an underscore '_'.

4.3 NSAP object class

An NEA, OSI or INTERNET network service access point in an end system or transfer system is represented by an NSAP object.

fssadm␣create␣NSAP
␣name={␣**nea-addr=**|␣**internet-addr=**|␣**osi-addr=**}[␣**net=**]

name=
Name of the NSAP object

1-32 printable visible characters

nea-addr=
NEA address in the form:

Process number/region number

Possible values for processor numbers and region numbers:

0...255

internet-addr=
Internet address in the form:

number.number.number.number

number
Decimal number 0...255

osi-addr=
OSI address in the Reference Publication Format according to IS 8348 Add2

net=
Type of network which is used by the local system to reach the NSAP.

N[EA]

I[NTERNET]

OSI-CO[NS]

snpa-list=

List of SNPAROUTES objects which can be used as an alternative way of reaching this NSAP in the form:

snpa[/weight][+snpa[/weight]]...

snpa

Name of an SNPAROUTES object

1-15 characters: letters, digits, the characters '_' and '#'.
Upper case and lower case are distinguished. The first character must not be a digit or an underscore '_'.

weight

Specification of a priority for the routes included in the list.

1...20

20 is the highest priority. The SNPAROUTES object specified in the list with *snpa*/20 is used as the first alternative route.

The list can have a maximum of 20 entries.

4.4 SUBNET object class

Objects of the SUBNET class represent a local subnet connection, which is uniquely identified by a subnet ID, or a group of local subnet connections, which is identified by the subnet ID that is common to these connections (*subnet* attribute).

The object is assigned values which are required to set X.25 minimum integration for calls with unknown partners and to activate and deactivate access control.

fssadm␣create␣SUBNET
subnet=[␣**incoming-call=**][␣**x25-description=**]
[␣**osi-nsap-address=**]

subnet=
Subnet ID

ISDN-i
i=1...32
For subnet access via ISDN switched connections (with or without the X.25 protocol) or via ISDN permanent connections (without the X.25 protocol)

X25-i
i=1...32
For subnet access via ISDN switched connections with the X.25 protocol

[incoming-call=]
Together with the *admit* attribute (FACIL object class), this attribute provides the configuration function for access control in CCP-ISDN. It acts as a switch for activating and deactivating access checks (temporarily or permanently).

Only relevant to switched connections (ISDN switched connections and X.25-SVC), i.e. for the connection types ISDN, X31-MSA, X.25, and X31-SVC.

NONE
All incoming connection requests are rejected. Any *admit* attribute configured for the calling address is ignored. The subnet address test is deactivated.

RESTRICTED

Incoming connection requests are accepted only if an incoming call is configured as permissible for the calling address, i.e. if the corresponding SNPAROUTES object is assigned a FACIL object which has the attribute *admit=BOTH_IN_AND_OUT* or *admit=INCOMING_ONLY*. The subnet address test is activated.

ALL

All incoming connection requests are processed. Any *admit* attribute configured for the calling address is ignored. The subnet address test is deactivated.

[x25-description=]

The *x25-description* attribute is permitted only for ISDN subnet IDs.

It refers to the predefined description of the X.25 access in the configuration file.

This attribute sets X.25 minimum integration for ISDN calls from unknown partners and selects the specified description of the X.25 access. For ISDN calls from known partners, the X.25 minimum integration is set by configuring an SNPAROUTES object of the type X31-MSA which corresponds to the calling ISDN number. The selection of the X.25 access description is configured only by assigning a FACIL object with the attribute *x25-description* to this SNPAROUTES object. The SUBNET attribute *x25-description* therefore has no effect on known partners.

The value must match the value of the *DTE-NAME* operand in the configuration file.

Possible values:

1...8 characters: letters, digits, characters '$', '#' and '@'.
The first character must not be a digit. No distinction is made between upper and lower case.

[osi-nsap-address=]

OSI address in Reference Publication Format in accordance with IS 8348

Add2

This attribute is only permitted for objects with subnet ID X25-n (n = 1, 2,..., 32) and can only be specified in the *fssadm* commands *create*, *get* and *set*.

Any syntactically correct OSI-NSAP address is accepted. The address does not have to be unique: the same OSI-NASP may occur for different SUBNET objects and/or NSAP objects and/or the LOCNSAP object.

5 Configuring for different uses

This chapter contains instructions for configuring different connections: to NEA networks, OSI and TCP/IP networks and into the Internet. Connections to X.25 partners are also handled.

In addition to setting up the local ISDN access in the configuration file (CF), complete configuration also encompasses definition of the accessible partners in the Transport Name Service (TNS) and Forwarding Support Service (FSS). The “CMX, Operation and Administration” User Guide [1] provides you with an overview.

You are shown an example configuration for each network interface together with the necessary configuration parameters and entries in the corresponding menus: for local ISDN access configuration as well as for the TNS and FSS. You are then shown corresponding extracts from TNS and FSS files.

You should be familiar with the general structure and operation of the CMX menu. The following instructions do not go into this in detail. Information can be found in the chapter “Operating CCP-ISDN” on page 47 and the “CMX, Operation and Administration” User Guide [1].

5.1 Preparatory checklists

This section contains the following information to provide you with a better overview while configuring your connections:

– summary of the various connections together with the possible protocol combinations

– checklist for the necessary configuration parameters

– information on the various subscriber numbers and network providers

5.1.1 Network connections, protocol combinations and CCP profiles

The table provides you with a summary of the possible protocol combinations and a selection also is shown in the relevant menus. The corresponding CCP profiles, which are accessed in CMX during TS application definition (TNS), are also shown.

Network interface	**Transport protocol**	**Network protocol**	**CCP profile**
NEA	NEATE/NEAN	HDLC X.25	ISDN-NEA ISDN-NX25
OSI	ISO8073	T.70 X.25	ISDN-CONS ISDN-CONS
TCP/IP	TCP/IP	T.70 HDLC PPP X.25	ISDN-HDLC ISDN-HDLC ISDN-HDLC ISDN-X25
X.25	OTHERS 1)	X.25	ISDN-X25 ISDN-X25

Table 7: Network interfaces with possible protocol combinations and CCP profiles
1) Private transport protocols

All connections to an X.25 network are handled in section “X.25 connections via ISDN”, regardless of the transport protocol used.

5.1.2 Configuration parameters required

The remote partner addresses and applications in the TNS and FSS must be set during configuration, as well as the local connection in the configuration file.

5.1.2.1 Configuration parameters for the local ISDN access

The following entries are possible when you configure your local ISDN connection:

– access mode (switched or permanent)

– configuration (bus or point-to-point)

– signaling (DSS1 or 1TR6)

– local ISDN subscriber number(s) for switched connections

– access/channel for permanent connections

– subnetwork ID

– X.25 connection description (name and local X.25 DTE address, facilities)

5.1.2.2 Configuration parameters for the TNS and FSS

Subnetwork addresses, routes, network addresses and local and remote applications are defined and administered in the TNS and FSS. The following parameters must be input:

– subnetwork type

– subnetwork address for a switched connection: remote ISDN subscriber number

– subnetwork address for a permanent connection: CC and line number (table 22)

	1st S_0 access	**2nd S_0 access**
Permanent connection	**00** for D-channel **01** for 1st B-channel **02** for 2nd B-channel	**32** for D-channel **33** for 1st B-channel **34** for 2nd B-channel

Table 8: Line numbers (decimal) for permanent connections

– network type

– network address, dependent on network type (table 9)

Network	**NEA**	**OSI**	**TCP/IP**	**X.25**
Format	Processor/ region	Remote ISDN subscriber number	Internet address	Remote X.25 DTE address
Example	1/100	0894656260	88.11.04.14	45890010123

Table 9: Format of the network addresses, with example

– name of the remote computer

– name of the local and remote applications

– CCP profile (see table 7)

– for X.25 connections: remote DTE address

FSS entries are made in the CMX main menu. You can, however, also make these entries directly in a configuration file with a text editor or by means of the Command Line Interface.

5.2 NEA connections

You can set up a connection to NEA networks via your computer connections using CCP-ISDN. You must first configure your own ISDN access and make entries in the TNS and FSS.

The following configuration is to be made:

Switched connections for a file transfer application are to be established to a TRANSDATA computer via an S_0 connection. The network provider has provided the S_0 connection in a bus configuration, and it has three subscriber numbers (101, 102 and 103).

5.2.1 Entries for own ISDN subnet connection

1. Select as the network access: ISDNS0
2. Create a configuration file and give the configuration file a meaningful name, e.g. *nea1.kogs*.
3. Define the following connection properties for connection 1:

   Access mode: Switched connection with DSS1 and BUS configuration.
4. Select a subnetwork ID (e.g. ISDN-1) and set “X.25 integration” to NO.
5. Enter the ISDN subscriber numbers that the network provider provided for your connection (101, 102 and 103)
6. Compile the configuration file.

The configuration of the ISDN connection is now completed.

5.2.2 Entering the addresses in the TNS and FSS

NEA connections require address entries in the FSS as well as the entries in the TNS. NEA partners use the NEATE transport protocol. The NEA address is defined in the TNS and the route over which the NEA partners are reached, is described in the FSS.

TNS entries

You must now configure the transport applications, file transfer in this case. You define both the local and remote applications.

Configure your local file transfer application first.

1. Input name part 5 for the local application in the

   GLOBAL NAME of TS application in DIR1:
   $FJAM_OUTBOUND
   for active connection setup of a file transfer application.
   $FJAM_OUTBOUND
   permits parallel processing of 100 jobs with *open*FTpartners.

2. Make the following entries for the LOCAL NAME:

CCP profile:	ISDN-NEA
T selector:	Input the format indicator and the name of the local application: *T* for TRANSDATA format and $FJAMOUT for the local file transfer application.

Now configure the remote file transfer application.

1. Input name part 5 for the remote application in the

   GLOBAL NAME of TS application in DIR1*:*
   ftisdnea for the remote file transfer application

2. Assign the TRANSDATA transport address with the following inputs:

CCP profile:	ISDN-NEA
Station name of TS application:	*T* for TRANSDATA format and *$JFAM* for the remote file transfer application.
Name of the remote system:	name of the NSAP entry from the FSS, e.g. *xyz*.
TRANSDATA network address:	processor/region number that you have obtained from your remote partner, e.g. 1/100.
CC list:	NO

FSS entries

You can find a description of the FSS parameters in chapter “FSS configuration” on page 65.

First define the route (object SNPA), then the remote computer (object NSAP). Input the commands with their attributes consecutively, separated with blanks.

1. You can only make entries in the FSS as the system administrator. Input the system administrator password.
2. Make the entries for SNPA (routes):

   ```
   fssadm create SNPAROUTES name=neaweg1 subnet=isdn-1
   type=ISDN-isdn-nr=<ISDN subscriber number of your partner>
   ```

   The entries have the following meaning:

   `name=neaweg1`
   Freely selectable name of the route.

   `subnet=isdn-1`
   Subnetwork ID that you assigned when you configured the local ISDN access, in this case ISDN-1.

   `type=ISDN`
   Type of subnetwork address.

   `isdn-nr=<ISDN subscriber number of your partner>`
   Here you specify the ISDN subscriber number of your NEA partner computer including the local access code (e.g. 04012345).

3. Now make the entries for the NSAP (remote computer):

   ```
   fssadm create NSAP name=xyz nea-addr=1/100 snpa-list=neaweg1
   ```

   The entries have the following meaning:

   `name=xyz`
   Name of the remote end system, a TRANSDATA computer.

   `nea-addr=1/100`
   Type of address (NEA) and number (processor/region), that you must obtain from your remote partner.

   `snpa-list=neaweg1`
   List of routes (SNPA objects); input the name assigned under Route.

This completes the FSS entries. You can check your FSS entries with the following command:

```
fssadm get SNPAROUTES neaweg1
fssadm get NSAP xyz
```

The NEA network interface is now completely configured.

5.2.3 Configuration files for NEA connection

FSS entries

```
SNPAROUTES (name=neaweg1 subnet=ISDN-1 type=ISDN-isdn-nr=
04012345)

NSAP (name=xyz nea-addr=1/100 net=NEA access=DIRECT
snpa-list=neaweg1)
```

TNS entries

```
$FJAM_OUTBOUND\
TSEL  WANNEA   T‘$FJAMOUT‘  ; application name:
ftisdnea
TA    WANNEA  (
      T‘$FJAM‘     ; station name
      1/100    )     ; remote NEA address: processor/region
```

5.3 OSI connections

You can set up connections from your computer to OSI networks with CCP-ISDN. You must, however, first configure your own ISDN access and make entries in the TNS for the remote partner. OSI partners use the ISO 8073 transport protocol and are only defined in the TNS.

The following configuration is to be defined: the network provider has established your S_0 connection as a permanent connection to your OSI partner and has provided one B-channel for your use. The application should be a terminal emulation.

5.3.1 Entering your own ISDN subnetwork interface

1. Select as the network access: ISDNS0.
2. Create the configuration file: give your configuration file a meaningful name, e.g. *osi1.kogs*.
3. Select the following connection property for connection 1: “Permanent connection/B”.
4. Configure the channel. Assign a subnetwork ID to the channel (e.g. ISDN-2). Select “TP02/T.70” as the protocol for all the channels.

   Compile the configuration file.

The ISDN access is now completely configured.

5.3.2 Entering the addresses in the TNS

The remote partner for OSI connections is defined exclusively in the TNS. You must also enter the local and remote application and the remote partner. You must define the local and remote application and the remote partner.

Configure your local application first.

1. Input name part 5 for the local application in the GLOBAL NAME of TS application in DIR1:
   osilokal for the active connection setup of your application.
2. Make the following entries for the LOCAL NAME:

CCP profile:	ISDN-CONS
T selector:	Input the format indicator and the name of the local application: *A* for ASCII format and *emulokal* for the local file transfer application.

Now configure the remote application and remote partner.

1. Input name part 5 for the remote application in the GLOBAL NAME of TS application in DIR1*:osifern* for the remote application
2. Assign the TRANSDATA transport address with the following inputs:

CCP profile:	ISDN-CONS
T selector of the TS application:	*A* for ASCII format and *emufern* for the remote application.
Type of subnet address:	Dedicated Line, Permanent Connection
CC list	YES
Transport protocol class	0/-
WAN CCs and lines	W1:1

The OSI network interface is now completely configured.

5.3.3 Configuration files for an OSI connection

TNS entries

```
osilokal\
   TSEL  WANSBKA  A‘emulokal‘  ; application name
osifern\
   TA  WANSBKA
   A‘emufern‘                  ; application name
   0/-                         ; transport protocol class
   WAN 1:1                       ; number of CC, 1st B-channel
```

5.4 TCP/IP with PPP via ISDN

You can set up connections from your computer to the TCP/IP network or the Internet using CCP-ISDN. You must configure your own ISDN access and make route entries in the FSS. Further information can be found in the “CMX, TCP/IP via WAN/ISDN” User Guide [3].

The following configuration is to be made: a CC with two S_0 connections using DSS1 signaling. Both S_0 connections are to create a “multiport” and can therefore be reached using a single subscriber number.

5.4.1 Entering your own ISDN subnetwork interface

1. Select as the network access: ISDNS0
2. Create the configuration file: give your configuration file a meaningful name, e.g. *ip1.kogs*.
3. Define the following connection properties for both connections:

Access mode:	SWITCHED CONNECTION
Signaling:	DSS1
Configuration:	PZP

4. Specify the following for both connections:

Subnet ID:	Assign an unused subnet ID, e.g. *ISDN-7*.
X.25 integration:	NO (See section “X.25 connections via ISDN” on page 105)

5. Configure the same ISDN subscriber number (MSN), which you get from the network provider, for both connections (e.g. 4711).

   Compile the configuration file.

Both S_0 connections are now completely configured.

5.4.2 Entering remote addresses in the FSS

You must enter your remote partner in the FSS for socket applications using ISDN. The TCP/IP transport protocol is used. Further information can be found in the “CMX, TCP/IP via WAN/ISDN” [3] User Guide.

First define the route (object SNPAROUTES) and then the remote computer (object NSAP). Enter the commands with their attributes consecutively, separated by blanks.

1. Make the following facility entry to use the protocol (SNP) PPP:

   ```
   fssadm create FACIL name=PPP ppp-profile=STANDARD
   ```

2. Make the entries for SNPAROUTES (routes):

   ```
   fssadm create SNPAROUTES name=tcpweg1 subnet=isdn-7
                  isdn-nr=040567788 facil=PPP
   ```

   The entries have the following meaning:

   `name=tcpweg1`
   : Freely selectable name of the route.

   `subnet=isdn-7`
   : Subnetwork ID that you assigned during configuration of the local ISDN access, in this case *isdn-7*.

   `isdn-nr=...`
   : The remote ISDN subscriber number must be input together with the complete local access code for switched connections.

   `facil=PPP`
   : Reference to the facility entry called PPP, which defines PPP routing.

3. Now make the entries for NSAP (remote computer):

   ```
   fssadm create NSAP name=abc internet-addr=88.66.55.10
                snpa-list=tcpweg1
   ...
   ```

   The entries have the following meaning:

   `name=abc`
   : Name of the remote end system (NSAP).

   `internet-addr=...`
   : Type and number of the address that you must obtain from your remote partner. The Internet address is required for TCP/IP.

`snpa-list=tcpweg1`
: List of routes (SNPAROUTES objects); enter the name you assigned for Route here.

The FSS entries are now completely configured. You can check your FSS entries with the following command:

```
fssadm get SNPAROUTES tcpweg1
fssadm get NSAP abc
fssadm get FACIL PPP
```

5.4.3 Configuration files for TCP/IP via ISDN

FSS entries

```
FACIL (name=PPP ppp-profile=STANDARD)

SNPAROUTES (name=tcpweg1 subnet=ISDN-7 type=ISDN
   isdn-nr=040567788 facil=PPP)

NSAP (name=abc internet-addr=88.66.55.10 net=INTERNET
   access=DIRECT snpa-list=tcpweg1)
```

5.5 Configuration files for TCP/IP partners in the GSM network (mobile users)

You can also use your ISDN connection to reach mobile partners, who use a laptop and cell telephone to connect to ISDN via the GSM network.

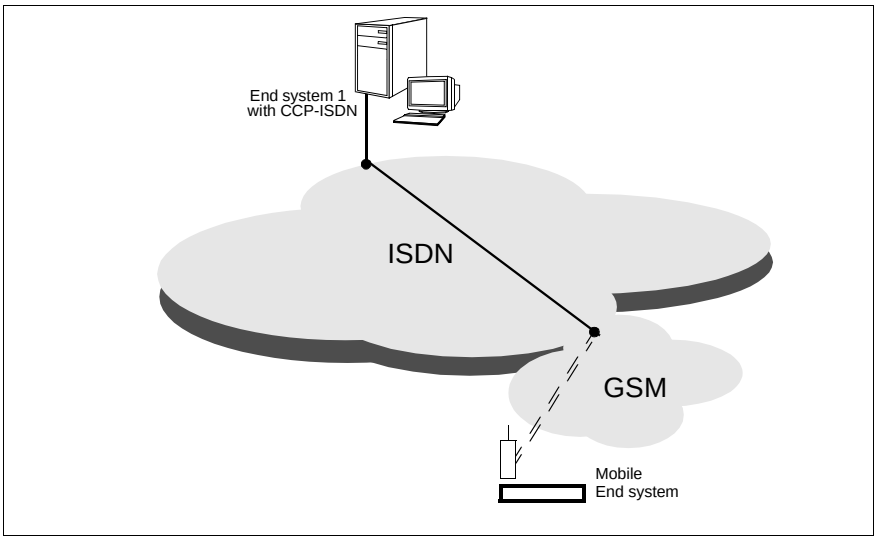


Figure 13: TCP/IP partners connected to GSM network

Entries for own ISDN connection

You can use the entries in section “Entering your own ISDN subnetwork interface” on page 100 for your ISDN connection.

Entering the addresses in the FSS

As a special feature in the mobile telephone network two subscriber numbers must be entered during partner configuration. As target subscriber number to the mobile telephone network user you enter the “number data 9.6” from the cell telephone user which you get from the network provider (SNPAROUTES “Partnr_Out”). For calls received from users of the mobile telephone network user you enter the “number telephony” of the cell telephone user which you get from the network provider (SNPAROUTES “Partner_In”).

1. Make the following facility entries to use the GSM network:

```
fssadm create FACIL (name=D2_IN ppp-Profile=GSM ppp-accm="DC1+DC3"
admit=INCOMING_ONLY)
fssadm create FACIL (name=D2_OUT ppp-profile=GSM ppp-accm="DC1+DC3"
admit=OUTGOING_ONLY)
```

2. Make the entries for SNPAROUTES:

```
fssadm create SNPAROUTES name=PARTNER_OUT subnet=isdn-7
isdn-nr=<number-data> facil=D2_OUT
fssadm create SNPAROUTES name=PARTNER_IN subnet=isdn-7
isdn-nr=<number-telephone> facil=D2_IN
```

3. Make the entries for the subscribers:

```
fssadm create NSAP name=abc internet-addr=88.66.55.10 snpa-list=
PARTNER_OUT+PARTNER_IN
```

FSS file for GSM

Once you have made the entries described above the FSS entries look like this:

FSS entries

```
FACIL (name=D2_IN ppp-profile=GSM isdn-ra=V110-ASYN
isdn-throughput=9,6 admit=INCOMING_ONLY ppp-accm="DC1+DC3")

FACIL (name=D2_OUT ppp-profile=GSM isdn-ra=V110-ASYN
isdn-throughput=9,6 admit=OUTGOING_ONLY ppp-accm="DC1+DC3")

SNPAROUTES (name=PARTNER_OUT subnet=ISDN-7 type=ISDN
isdn-nr=<number> facil=D2_OUT)

SNPAROUTES (name=PARTNER_IN subnet=ISDN-7 type=ISDN
isdn-nr=<number> facil=D2_IN)

NSAP (name=abc internet-addr=88.66.55.10 access=DIRECT
snpa-list="PARTNER_OUT+PARTNER_IN")
```

5.6 X.25 connections via ISDN

This section deals with configuring ISDN connections to an X.25 network.

You make the entries for your remote partner in the TNS/FSS. Partners that interface directly with the X.25 protocol and not via NEA, IP or TP02, use OTHERS. The remote partners of these applications are defined in the TNS.

The following configuration is described in this section:

- X.25 minimum integration for the S_0 connection as OSI connection
- X.25 maximum integration for the S_0 connection with X.25 in the D-channel without transport protocol
- X.25 maximum integration for the S_{2M} connection with X.25 in the B-channel as NEA connection

5.6.1 X.25 minimum integration for S_0 connection as an OSI connection

5.6.1.1 Entries for your own ISDN subnetwork interface

The following information is required for configuring the X.25 minimum integration:

– your own local subscriber number(s) of the ISDN connection

– the DTE name

  You should set the DTE name for all X.25- connections via ISDN. You must specify it if two or more X.25 network accesses are configured in the corresponding KOGS by means of the XZSTW-KOGS macro. If only one X.25 network access is configured in the KOGS by means of XZSTW then you can omit this specification since this X.25 network access is used as the default.

– the local DTE address

  This is used as the calling address during X.25 connection setup and consists of decimal digits.

– channel ranges for SVC

– transfer route

– X.25 facilities

You must obtain the required settings from your X.25 network provider or network administrator. See section “X.25 facilities” on page 22 for more information.

Configuring a subnet connection

1. Select as the network access: ISDNS0
2. Create the configuration file: give your configuration file a meaningful name, e.g. *x25.kogs*.
3. Define the following connection properties:

Access mode:	SWITCHED CONNECTION
Signaling:	DSS1
Configuration:	BUS

4. Specify the following for the connection:

Subnet ID:	Assign an unused subnet, e.g. *ISDN-5*.
X.25 integration:	YES

5. Specify the following for the ISDN subscriber number:

Subscriber number:	z. B. 771122

6. Enter a name for the X.25 connection description. The name must agree with the name specified in FSS.

DTE name:	e.g. x25#1
X.25 integration:	minimum

7. Enter the following for the specified X.25 DTE:

DTE address:	Obtain your local DTE address from the X.25 network provider, e.g. 45890010123.
Channel ranges:	Enter the proposed channel ranges as agreed with the X.25 network provider.
X.25 variant:	Specify which part your connection takes on for X.25 communication, in this case DTE, and which X.25 variant you wish to use, e.g.: DTE_CCITT_1984
Agreed X.25 facilities	NO

8. Compile the configuration file.

The configuration of the ISDN connection is now complete.

5.6.1.2 Entering the addresses in TNS and FSS

Entries in TNS

1. Make the following entries in the local application.

```
x25lokal1\
   TSEL WAN3SBKA A‘x29app01‘
```

   The entries have the following meanings:

   `x25lokal1`
   : global name for active connection setup of your application

   `TSEL WANSBKA`
   : address format of the transport address, in this case a transport protocol in accordance with IS 8073 (OSI).

   `A‘x25lokal1‘`
   : Format indicator and name of the local application:
   A for ASCII format and *x25lokal1* for the application.
   X.29 applications must register with X29APPnn in the system, whereby nn stands for two decimal digits from 00 to 99.

2. Make the following entries for the remote application.

```
x251fern1
   TA WAN3SBKA X.31 07089654 X.121 45890010777
   A‘x25fern1‘
```

   The entries have the following meaning:

   `x251fern1`
   : global name for remote application

   `TA WANSBKA`
   : address format of the transport address, in this case in this case a transport protocol in accordance with IS 8073 (OSI).

   `A‘x25fern1‘`
   : T selector and name of the remote application:
   A for ASCII format and *x25fern1* for the remote application.

   `X.31 07089654`
   : After the address identification X.31 enter the remote ISDN subscriber number of the Interworking Unit with complete local access number.

`X.121 45890010777`
: Enter the DTE address of the end system of your X.25 partner after the address identification X.121. If a transfer is made to a X.25 network (not therefore in a DTE-DTE link), the international dialing prefix of the X.25 network must be specified before the DTE address of the partner (for the Datex-P network of Deutsche Telekom it is 0262).

3. Save the file and exit it.
4. Enter the following command with which the information is passed on to the TS directories.

```
tnsxcom -u <file name>
```

FSS entries

1. Description of the X.25 access

```
fssadm create FACIL name=x2501 x25-description=x25#1
npid=OSI-CONS
```

The entries have the following meaning:

`name=x2501`
: name referred to by SNPAROUTES.

`x25-description=x25#1`
: refers to the predefined description of the X.25 access in the CC configuration file (DTE name).

`npid=OSI-CONS`
: ID of the TP0/2 network protocol.

2. Define the route to the remote system

```
fssadm create SNPAROUTES name=x25min1 subnet=isdn-5
x31-msa=07089654/45890010777 facil=x2501
```

The entries have the following meaning:

`name=x25min1`
: freely selectable name of the route

`subnet=isdn-5`
: subnet ID which you have issued during the configuration of the local ISDN connection.

`x31-msa=07089654/45890010777`
: Two-step dialing address for a connection according to X.25 minimum integration: ISDN number/DTE address. If a transfer is made to a X.25 network (not therefore in a DTE-DTE link), the international dialing prefix of the X.25 network must be specified before the DTE address of the partner (for the Datex-P network of Deutsche Telekom it is 0262).

`facil=x2501`
: refers to the FACIL object with the name *x2501*.

The X.25 network connection is now fully configured.

5.6.1.3 Configuration files for X.25 connection via ISDN

TNS entries

The ISDN subscriber number address identification is X.31. The DTE address identification is X.121.

```
x25lokal1\
TSEL  WANSBKA  A‘x25lokal1‘  ; name of application
x25fern1\
TA  WANSBKA  (
    X.31
    07089654      ; remote ISDN subscriber number of IU
    X.121
    45890010777   ; remote DTE address of X.25 partner
    A‘x25fern1‘ ) ; name of application
    WAN 1   )
```

FSS entries

```
FACIL ( name=x2501 x25-description=x25#1 npid=OSI-CONS )

SNPAROUTES ( name=x25min1 subnet=isdn-5 type=X31-MSA
x31-msa=07089654/45890010777 facil=x2501)
```

5.6.2 X.25 maximum integration for S_0 connection in the D-channel without transport protocol

The following sample configuration shows an X.25 maximum configuration for a S_0 connection in the D-channel with a private application based on the X.25 protocol.

5.6.2.1 Entries for your own ISDN subnet connection

1. Select as the network access: ISDNS0
2. Create the configuration file: give your configuration file a meaningful name, e.g. *x25d.kogs*.
3. Define the following connection properties:

Type of connection:	SWITCHED CONNECTION
Signaling:	DSS1
Configuration:	BUS

4. Specify the following for the connection:

Subnet ID:	Assign an unused subnet, e.g. *ISDN-1*.
X.25 integration:	YES

5. Specify the following for the ISDN subscriber number:

Subscriber number	e.g. 4411

Configuring parameters for X.25 integration

1. Enter a name for the X.25 connection description. The name must agree with the name specified in FSS.

DTE name:	e.g. x25maxd
X.25 integration:	maximum/D

2. Make the following entries for the specified X.25 DTE:

DTE address:	Obtain your local DTE address from the X.25 network provider, e.g. 774411.
Channel ranges:	Enter the proposed channel ranges as agreed with the X.25 network provider.
TEI:	Enter the Terminal Endpoint Identifier assigned by the network provider: e.g. 1
X.25 variant:	Enter which role your connection takes on for the X.25 communication, in this case DTE, and which X.25 variant you wish to use, e.g.: DTE_CCITT_1984
Agreed X.25 facilities:	NO

3. Compile the configuration file.

The configuration of the ISDN connection is now completed.

5.6.2.2 Entering the addresses in TNS and FSS

Entries in TNS

1. Make the following entries for the local application.

```
x25lokal1\
TSEL WAN3SBKA A‘noux2902‘
```

The entries have the following meanings:

`x25lokal1\`
: global name for active connection setup of your application

`TSEL WAN3SBKA`
: address format of the network access for X.25.

`A‘noux2902‘`
: Format indicator and name of the local application:
A for ASCII format and *noux2902* for the application privat x.25.

2. Make the following entries for the remote application.

```
x25fern1\
   TA WAN3SBKA X.121 4589668822
   A‘x25fernd’
```

   The entries have the following meaning:

   `x251fern1\`
   : global name for remote application

   `TA WAN3SBKA`
   : address format of the network access for X.25.

   `A‘x25fernd\‘`
   : T selector and name of the remote application:
   A for ASCII format and *x25fernd* for the remote application.

   `X.121 4589668822`
   : enter the DTE address of the end system of your X.25 partner after the address identification X.121. If a transfer is made to a X.25 network (not therefore in a DTE-DTE link), the international dialing prefix of the X.25 network must be specified before the DTE address of the partner (for the Datex-P network of Deutsche Telekom it is 0262).

   `WAN 1`
   : Number of the CC.

3. Save the file and exit.

4. Enter the following command with which the information is forwarded to the TS directories.

```
tnsxcom –u <file name>
```

FSS entries

1. Description of the X.25 access

```
fssadm create FACIL name=x25d
                x25-description=x25maxd
```

   The entries have the following meaning:

   `name=x25d`
   : name referred to by SNPAROUTES.

   `x25-description=x25maxd`
   : refers to the predefined description of the X.25 access in the CC configuration file (DTE name).

2. Define the route to the remote system

```
fssadm create SNPAROUTES name=x25d subnet=isdn-1
               x31-dte-addr=4589668822 facil=x25d
```

   The entries have the following meaning:

   `name=x25d`
   : freely selectable name of the route

   `subnet=isdn-1`
   : subnet ID which you have issued during the configuration of the local ISDN connection.

   `x31-dte-addr=4589668822`
   : remote address of partner. If a transfer is made to a X.25 network (not therefore in a DTE-DTE link), the international dialing prefix of the X.25 network must be specified before the DTE address of the partner (for the Datex-P network of Deutsche Telekom it is 0262).

   `facil=x2501`
   : refers to the FACIL object with the name *x25d*.

The X.25 network connection is now fully configured.

5.6.2.3 Configuration files for X.25 connection via ISDN

TNS entries

```
x25lokal1\
    TSEL  WAN3SBKA  A‘noux2902‘  ; application name
x25fern1\
    TA  WAN3SBKA  ( X.121
    4589668822     ; remote DTE address of X.25 partner
    A‘x25fernd‘    ) ; application name
```

FSS entries

```
FACIL ( name=x25d x25-description=x25maxd )
SNPAROUTES ( name=x25d subnet=isdn-1 type=X31-SVC
             x31-dte-addr=4589668822 facil=x25d)
```

5.6.3 X.25 maximum integration for S_{2M} access with X.25 in B-channel as an NEA access

5.6.3.1 Entries for your local ISDN subnet access

1. Select as the network access: ISDNS2.
2. Create the configuration file: give your configuration file a meaningful name, e.g. *x25b.kogs*.
3. Specify the following for the connection:

Subnet ID:	Assign an unused subnet, e.g. *ISDN-2*.
X.25 integration:	YES

4. Specify the following for the ISDN subscriber number:

Subscriber number	e.g. 324252
User protocol:	NEA

Configuring parameters for X.25 integration

1. Enter a name for the X.25 connection description. The name must agree with the name specified in FSS.

DTE name:	e.g. x25maxb
X.25 integration:	maximum/B

2. Make the following entries for the specified X.25 DTE:

DTE address:	Obtain your local DTE address from the X.25 network provider, e.g. 324252.
Channel ranges:	Enter the proposed channel ranges as agreed with the X.25 network provider.
X.25 variant:	Enter which role your connection takes on for the X.25 communication, in this case DTE, and which X.25 variant you wish to use, e.g.: DTE_CCITT_1984
Agreed X.25 facilities:	NO

3. Compile the configuration file.

The configuration of the ISDN connection has now been completed.

5.6.3.2 Entries the addresses in TNS and FSS

Entries in TNS

1. Make the following entries for the local application.

```
x25lokalb\
   TSEL WANNEA A‘x25lokalb‘
```

   The entries have the following meaning:

   `x25lokalb`
   global name for active connection setup of your application

   `TSEL WANNEA`
   address format of the transport address, in this case NEA.

   `A‘x25lokalb‘`
   format indicator and name of the local application:
   A for ASCII format and *x25lokalb* for the application.

2. Make the following entries for the remote application.

```
x25fernb\
TA WANNEA A‘x25fernb‘ 1/2
```

The entries have the following meaning:

`x25fernb`
: global name of the remote application

`TA WANNEA`
: address format of the transport address, in this case NEA.

`A‘x25fernb‘`
: T selector and name of the remote application:
A for ASCII format and *x25fernb* for the remote application.

`1/2`
: NEA address in the form process number/region number

3. Save the file and exit.
4. Enter the following command with which the entries are accepted by the TS directories.

```
tnsxcom –u <file name>
```

The remote partners for the first X.25 application are now entered. Note that your NEA application must also forward X.25 call user data at connection setup.

FSS entries

1. Description of the X.25 access

```
fssadm create FACIL name=x25b npid=NEA
                x25-description=x25maxb
```

The entries have the following meaning:

`name=x25b`
: name of the facility object referred to by SNPAROUTES.

`npid=NEA`
: NEA network protocol code

`x25-description=x25maxb`
: refers to the predefined description for X.25 access in the CC configuration file (DTE name).

2. Define the route to the remote system

```
fssadm create SNPAROUTES name=x25b subnet=isdn-2
               x31-dte-addr=4589625242 facil=x25b
```

The entries have the following meaning:

`name=x25b`
freely selectable name of the route

`subnet=isdn-2`
subnet ID which you have issued when configuring the local ISDN connection.

`x31-dte-addr=`4589625242
X.25 address of the partner (remote address). If a transfer is made to a X.25 network (not therefore in a DTE-DTE link), the international dialing prefix of the X.25 network must be specified before the DTE address of the partner (for the Datex-P network of Deutsche Telekom it is 0262).

facil=x25b
refers to the FACIL object with the name *x25b*.

3. Information on the remote system

```
fssadm create NSAP name=x25b nea-addr=1/2 snpa-list=x25b
```

The entries have the following meaning:

`name=x25b`
name of the NSAP object.

`nea-addr=1/2`
NEA address of the remote system

`snpa-list=x25b`
refers to an SNPAROUTES object, in which the route is defined.

The X.25 network connection is now completely configured.

5.6.3.3 Configuration files for X.25 connection via ISDN

TNS entries

```
x25lokalb\
TSEL  WANNEA  A‘x25lokalb‘ ; application name
x25fernb\
TA  WANNEA  (
    A‘x25fernb‘  ; application  name NEA address
    1/2   )
```

FSS entries

```
FACIL ( name=x25b npid=NEA x25-description=x25maxb )

SNPAROUTES ( name=x25b subnet=isdn-2 type=X31-SVC
   x31-dte-addr=4589625242 facil=x25b )

NSAP ( name=x25b nea-addr=1/2 snpa-list=x25b )
```

5.6.4 X.25 partner facilities in the FSS

X.25 facilities can be configured in the FSS for all remote network partners. This is independent of the transport/network profile used. In the SNPAROUTES object, the FACIL object is referred to using the attribute *facil* which contains the facilities. This has no influence on TNS or KOGS.

The attribute *x25-description* = ... is only relevant for X.25 minimum or maximum integration and refers to the description of the X.25 access in the configuration file (DTE name).

FSS entries

```
FACIL ( name=fa091512
       x25-packet-size=1024/1024 x25-window-size=7/7
       x25-cug=05 x25-revch=REQUEST_ONLY
        x25-fast-select=NO_RESTRICTION x25-description=X25#2 )
SNPAROUTES ( name=ro091616 subnet=X25-22 type=X25
              dte-addr=1600000 facil=fa091512 )
```

The possible FSS attributes for the object class FACIL are described in chapter chapter “FSS configuration” on page 65.

Special case Fast Select

The *Fast Select* facility must be requested from the network provider and configured appropriately:

FSS entries

In addition, an entry must be made in the FSB for each X.25 partner with which Fast Select is agreed:

```
FACIL ( name=fa091512 x25-fast-select=NO_RESTRICTION )
SNPAROUTES ( name=ro091616 subnet=X25-22 type=X25
             dte-addr=1600000 facil=fa091512 )
```

In the FACIL object it is specified that “Fast Select” is signaled for an outgoing call.

5.6.5 User-defined number of X.25 switched connections (SVC) via a B-channel

In the FSS parameter *x31min-svc-to-Bchan* you can determine how many X.25 SVCs are to be established to the same ISDN partner via one B-channel. If more SVCs than the maximum number are established, the next B-channel is used.

Significant only for DTE-DTE links or X.25 minimum integration.

FSS entries

```
FACIL ( name=fa1 x31min-svc-to-Bchan=5-TO-EACH )
```

In this example the first 5 SVCs use one B-channel. If more SVCs are set up, they use the next B-channel.

```
FACIL ( name=fa2 x31min-svc-to-Bchan=1-TO-EACH )
```

In this example each individual SVC uses a separate B-channel.

```
FACIL ( name=fa3 x31min-svc-to-Bchan=MAX-TO-EACH )
```

In this example of *MAX-TO-EACH*, SVCs continue to use a B-channel until the maximum number (defined in the X.25 description for the controllers, outgoing/incoming SVCs parameter) is reached. Additional SVCs are then set up via the next B-channel.

```
FACIL ( name=fa4 x31min-svc-to-Bchan=MAX-TO-ONLY-ONE )
```

In the default case of *MAX-TO-ONLY-ONE* all SVCs (up to a maximum number defined defined in the X.25 description for the controllers, outgoing/incoming SVCs parameter) set up to a partner use one B-channel. No other B-channel is set up.

The user-defined number of X.25 switched connections via a B-channel must not exceed the number of outgoing SVCs specified in the configuration file during the configuration of the X.25 access.

5.7 Access control

CCP-ISDN provides a system of scalable access control. Access control management is offered for a variety of connections, from the subnet access (subnet accesses can also be grouped) to the individual route.

Access control must be explicitly activated, as it is inactive by default. The incoming_call attribute in the SUBNET object class in the FSS is provided for this purpose (see section “SUBNET object class” on page 85).

5.7.1 Subnet ID-specific block for all incoming calls

FSS entry

```
fssadm create SUBNET subnet=ISDN-22
                incoming_call=NONE
```

`subnet=ISDN-22`
: Subnet ID *ISDN-22*

`incoming_call=NONE`
: Activates access control. All incoming switched connections for subnet ID *ISDN-22* are rejected.

5.7.2 Subnet ID-specific block for unknown callers

FSS entry

```
fssadm create SUBNET subnet=ISDN-22
               incoming_call=RESTRICTED
```

`subnet=ISDN-22`
: Subnet ID *ISDN-22*

`incoming_call=RESTRICTED`
: Activates access control. All incoming switched connections for subnet ID *ISDN-22* whose sender address is unknown are rejected.

  Only calls from partners whose address is entered correctly in the FSS and for whom the access control attribute *admit* has been set either with *INCOMING_ONLY* or with *BOTH_IN_AND_OUT* are accepted.

5.7.3 Access for known callers

If you have activated access control for your system with *incoming_call= RESTRICTED*, you must make the following entries for partners to whom you want to grant access:

FSS entries

1. Create a facility object with the access control attribute *admit*.

```
fssadm create FACIL name=access admit=INCOMING_ONLY
```

`name=access`
: Name of the FACIL object

`admit=INCOMING_ONLY`
: Only incoming calls are accepted

  If you also want an outgoing connection to partners who are to have access, use the value *BOTH_IN_AND_OUT* instead of *INCOMING_ONLY* or create an additional FACIL object for *BOTH_IN_AND_OUT*.

2. Then assign the FACIL object to the route.

```
fssadm create SNPAROUTES name=partner subnet=ISDN-22
                isdn-nr=089123456 facil=access
```

`name=partner`
: Name of the SNPAROUTES object (of the route)

`subnet=ISDN-22`
: Subnet ID *ISDN-22*. (You assigned this value when configuring the local ISDN connection.)

`isdn-nr=089123456`
: ISDN subscriber number of the partner. In switched connections the remote ISDN subscriber number is specified in full form, i.e. with the local access code.

`facil=access`
: Reference to *access*, the previously created FACIL object. This explicitly grants access to the partner with the subscriber number 089123456.

  Outgoing calls to the partner are not permitted, since *admit* is set to *INCOMING_ONLY*.

5.7.4 Checking the access control status

You can query the status of the access control at any time.

```
fssadm get SUBNET
```

lists all the managed SUBNET objects. Access control is active only if an object contains an *incoming_call* attribute set to *NONE* or *RESTRICTED*.

If a SUBNET object (e.g. ISDN-22) is entered with *incoming_call=RESTRICTED*, you can query the routes/partners for which access is permitted. The following commands are required for this:

```
fssadm get FACIL
```

and

```
fssadm get SNPAROUTES subnet=ISDN-22.
```

5.7.5 Deactivating access control

You can deactivate access control at any time by simply modifying the *incoming-call* attribute of the corresponding SUBNET object.

Example 1

```
fssadm set SUBNET subnet=ISDN-22 incoming-call=
```

The above command deletes the *incoming-call* attribute of this SUBNET object. All incoming calls are accepted - except in combination with CS-ROUTE, when the *admit* attribute is set to *OUTGOING_ONLY* or *NEITHER_IN_NOR_OUT*.

Example 2: FSS entry

```
fssadm set SUBNET subnet=ISDN-22 incoming-call=ALL
```

This command means that all incoming calls are accepted for the SUBNET object ISDN-22. The *admit* attribute is no longer evaluated for any route.

5.8 Configuration option for unknown partners for X.25 minimum integration

With CCP-ISDN you can configure settings for unknown callers with X.25 minimum integration. These settings can be used to simplify the management of mass data or to grant defined access to unknown partners.

The configuration option for unknown callers with X.25 minimum integration is valid only when access control is inactive.

This configuration has no effect on known partners nor in particular on outgoing calls.

Example: FSS entries

```
fssadm create SUBNET subnet=ISDN-1 x25-description=x25min
```

The created SUBNET object specifies that the subnet profile X.25 minimum integration is always to be set for all incoming calls from unknown partners at the subnet connection *ISDN-1*, and that the predefined definition of the X.25 access *x25min* is to apply. The currently loaded configuration file must therefore contain an X.25 access description called “x25min”, which is then referred to.

6 Administration and diagnosis commands for CC

This chapter describes the administration and diagnosis commands and the help functions for working with network access software and the Communication Controllers (CC).

You must administer and maintain subnet profiles and CCs in expert mode. The following steps are required for this (see “CMX, Operation and Administration” User Guide [1]):

1. Select *Communication Controller* in the CMX menu.
2. Select the CC you want to administer.
3. Select the *Enter expert mode* CC operation.

You are now in expert mode, as you can recognize by the displayed user interface character. It consists of the relevant CC identification and an asterisk (*), e.g. *W1**. You can input all administration commands in expert mode.

The response from the CC is time-monitored for commands that await an answer. An error message is output if the response does not arrive in time.

You can exit expert mode and return to the previous menu by pressing End or CTRL D.

Summary of commands

Below you will find a list of all commands, sorted according to task areas.

Task area	Command	Brief description
Administration commands: Test and change configuration	ach	Activate line
	assign	Assign a subnet profile to a CC
	compile	Compile configuration source file
	dah	Deactivate line
	exchange	Exchange configuration file
	info	Query CC state information
	linkstat	Output status of connections
	load	Load subnet profile
	stop	Stop subnet profile

Table 10: Summary of commands sorted according to task areas

Task area	Command	Brief description
Diagnostic commands: Trace list handling	dump format sof son tof ton	Dump CC memory Convert trace lists and dumps Switch trace off Switch trace on Switch trace list transfer off Switch trace list transfer on
Help functions	cmdfile : ? ! #	Execute command file Change administered CC List diagnostic mode commands Execute shell command Comment command file

Table 10: Summary of commands sorted according to task areas

Command format

The commands have the following format:

command[␣**-b**␣cc][...]

command
: Name of the command

Options
: An option consists of a hyphen, a selector and an argument. The hyphen and selector are printed bold in this manual.

  The blank between the selector and argument can be omitted.

  The options may be specified in any order.

  Options can be specified both in upper and lower case letters.

  -b␣cc
  : Specification of the CC ID. Changes the administered CC.

  This option occurs in almost all commands and is therefore explained at this point only, for all commands.

  cc
  : ID *W1*, *W2*, ... of the CC.

  -b␣cc not specified:
  : The CC you selected when you called diagnostic mode is preset.

[...]
Specification of further options.

Example of an administration command

Two options are available to you for changing the administered CC:

– Command entry with a colon (:).

  You are administering, e.g. CC *W1* and want to change to CC *W3*.

```
W1* : -b W3
W3*
.
.
```

– Command entry with the ID of the required CC.

  Input the required command together with *-b␣cc*.

  You are administering, e.g. CC *W2* and want to query information for CC *W3*.

```
W2* info -b W3
...
W2*
.
.
.
```

6.1 Administration commands

The administration commands are described in alphabetic order in the following section.

6.1.1 ach - activate line

The *ach* command activates a line configured in the KOGS, on a loaded CC. The software resources are provided and the local hardware is checked.

Line designates the complete ISDN connection for switched lines.

Line designates a single B-channel or the D-channel of an ISDN access for permanent connections.

ach␣[**-b**␣cc]␣**-l**␣n

-b␣cc
: Specification of the CC ID.

-l␣n
: Specification of the line number.

Connection	**1st S_0 or S_{2M} access**	**2nd S_0 access**
Switched	**00**	**32**
Permanent (only S_0)	**00** for D-channel **01** for 1st B-channel **02** for 2nd B-channel	**32** for D-channel **33** for 1st B-channel **34** for 2nd B-channel

Table 11: Line numbers (decimal) for switched and permanent connections

6.1.2 assign - assign network access software to a CC

The *assign* command is used to assign a network access software to a CC. This assignment only becomes effective the next time the CC is loaded.

assign[␣**-b**␣cc]␣**-c**␣ccp

-b␣cc
: Specification of the CC ID.

-c␣ccp
: Specification of the network access software that is to be assigned to the CC.

  ccp
  : Name of the network access software: **isdns0** or **isdns2**

    The current assignment is deleted if you input a minus sign instead of a name.

You can execute the *Change configuration* operation under the *Operations for CC* option (select the CC you want) in the *CCs - Communication Controller* CMX menu, instead of this command.

Example

You are administering CC *W1* and want to assign *isdns0* to CC *W2*.

```
W1* assign –b W2 –c isdns0
W1*
```

or

```
W1* : –b W2
W2* assign –c isdns0
W2*
```

6.1.3 compile - compile configuration source file

You use the *compile* command to call up a compiler that compiles the configuration source file for the subnet profile and CC. A user ID can be entered in the header of the compiled configuration file.

compile[␣**-b**␣cc]␣**-c**␣ccp␣**-f**␣file[␣**-u**␣userid]

-b␣cc
: Specification of the CC ID

-c␣ccp
: Specification of the network access software which is to be assigned to the CC.

  ccp
  : Name of the network access software: isdns0 or isdns2

-f␣file
: Specification of the name of the configuration source file that is assigned to the subnet profile and is to be compiled.

  file
  : File name. This may be a maximum of 10 characters long.

-u␣userid
: Specification of the user ID with which the configuration file can be identified. It is entered in the header of the compiled file.

  userid
  : The user identification may consist of four printable characters. Missing characters are extended using ASCII blanks.

6.1.4 dah - deactivate line

You deactivate a line on a loaded CC with the *dah* command.

Line designates the complete ISDN connection for a switched line.

Line designates a single B-channel or the D-channel of an ISDN access for permanent connections.

dah[␣**-b**␣cc]␣**-l**␣n

-b␣cc
Specification of the CC ID.

-l␣n
Specification of the line number.

Connection	**1st S_0 or S_{2M} access**	**2nd S_0 access**
Switched	**00**	**32**
Permanent (only S_0)	**00** for D-channel **01** for 1st B-channel **02** for 2nd B-channel	**32** for D-channel **33** for 1st B-channel **34** for 2nd B-channel

Table 12: Line numbers for switched and permanent connections

6.1.5 exchange - exchange configuration file

The configuration file (CF) for the network access software of the specified CC is exchanged or assigned with the *exchange* command.

exchange[␣**-b**␣cc]␣**-c**␣ccp␣**-k**␣file

-b␣cc
: Specification of the CC for which you want to exchange the CF.

-c␣ccp
: Specification of the network access software for which the CF is exchanged. It is also possible to execute exchange for a profile which isn't the currently assigned one.

  ccp
  : Name of the network access software: isdns0 or isdns2

-k␣file
: Name of the configuration that is to be assigned to the CC or network access software.

  file
  : Name of the configuration file. This may be a maximum of 10 characters.

    The extension *.ccp* is appended to the name within the system. The CF must first be created by compiling a KOGS file. The current assignment is deleted if a minus sign is specified instead of a name.

You can execute the *Change configuration* operation under the *Operations for CC* option (select the CC you want) in the *CCs - Communication Controller* CMX menu, instead of this command.

6.1.6 info - query CC state information

The *info* command is used to query the configuration of the CC and the state of the loaded subnet profile. Output is written to the screen.

info[␣**-b**␣cc]

-b␣cc

Specification of the CC ID for which the information is to be output.

-b␣cc not specified: information is output for all CCs.

Output of the info command

The information is output in the following format:

CC	State	Loaded network access software	Loaded configuration	Assigned network access software	Assigned configuration
W1	READY	CCP-ISDNS0	KOGS1.ccp	CCP-ISDNS0	KOGS2.ccp

The loaded configuration file (KOGS1.ccp) is currently relevant. The assigned configuration file (KOGS2.ccp) then becomes the currently relevant configuration when the CC is reloaded.

6.1.7 linkstat - Displaying status of the connection of CC

You can use the *linkstat* command to display the status of the connections of a particular CC.

linkstat␣**-b**␣cc[␣**-l**][␣**-h**]

-b␣cc
Specification of the CC ID.

-l
Specification that a detailed list should be output.

-h
Specification that no header should be output.

The output depends on the CCP used. The information is output as shown in the following table.

CC	IF#	STATE	TYPE	Bits/s	LINK	LINKS	...
W2	1/0	BUSY	S0	16k	LAPD	1/1	
W2	1/1	HWEN	S0	64k	–	0/1	
W2	1/2	HWEN	S0	64k	–	0/1	
W2	1/31	NETC	S0	9600	–	0/1	
W2	2/0	–	–	16k	–	–	
W2	2/1	–	–	64k	–	–	
W2	2/2	–	–	64k	–	–	
W2	2/31	–	–	–	–	–	

...	NETW.	SUBNET	SUBNET-ID	SUBNET-ADDR
	DSS1	BUS	ISDN_1	63019053
	–	BUS	ISDN_1	–
	–	BUS	ISDN_1	–
	–	BUS	ISDN_1	–
	–	–	–	–
	–	–	–	–

The individual columns are described in the following:

CC
: CC ID

IF#
: Connection consisting of connection number and ISDN channel in the form connection number/ISDN channel

STATE
: Status of the connection

-	Connection not configured
DISA	Connection disabled
LINK	Links are set up
BUSY	Maximum number of connections reached
HWEN	Hardware is ready for operation
NETC	Connection setup to network or node (e.g. dialing)

TYPE
: Physical connection type: S0 for S_0 access or S2 for S_{2M} connections

Bits/s
: Transfer speed

LINK
: Transfer protocol in the ISO data protection layer (layer 2)

LINKS
: Number of connection in the form:
existing connections / maximum number of connections

NETW.
: Type of network protocol used:
X.25, T70-3, PPP, DSS1, 1TR6

SUBNET

Physical connection is used as:

-	unknown
CSDN	Circuit Switched Digital Network
PHONE	Switched connection in telephone network
LEASED	Permanent connection
MP	Multipoint permanent connection
BUS	ISDN S_0 bus
PTP	ISDN point-to-point connection

SUBNET-ID

Displays the subnet identification of the respective ISDN connection defined in the configuration of the CC.

SUBNET-ADDR:

Displays the local subscriber number of the corresponding ISDN connection.

6.1.8 load - load network access software

You load the network access software together with the corresponding configuration (CF) onto a CC with the *load* command. The *load* command may not be called from within a command file.

The network access software that you previously assigned to the selected CC is loaded. The CF which was exchanged for the currently assigned network access software and CC now comes into effect.

load[␣**-b**␣cc]

-b␣cc
Specification of the CC ID.

You can execute the *Load CC...* operation under the *Operations for CC* option (select the CC you want) in the *CCs - Communication Controller* CMX menu, instead of this command.

If network access software is active when the load command is called, this software is stopped without warning and the currently assigned network access software is loaded.

6.1.9 stop - stop network access software

The *stop* command is used to deactivate loaded network access software and the loaded CC.

stop[␣**-b**␣cc]

-b␣cc
Specification of the CC ID

You can execute the *Unload CC* operation under the *Operations for CC* option (select the CC you want) in the *CCs - Communication Controller* CMX menu, instead of this command.

All existing connections via this CC are broken off. You should therefore ensure that no connections exist via this CC before calling this command.

6.2 Diagnostic commands

6.2.1 dump - dump CC memory

The *dump* command is used to write the current contents of the CC memory (program and data) into a file. This file contains important information for diagnosing errors. You can convert this file with the *format* command. The network access software must be reloaded onto the CC after dumping.

dump[␣**-b**␣cc][␣**-f**␣file]

-b␣cc
: Specification of the CC ID.

-f␣file
: Name of the file to which the required dump is to be written.

  The file name can be a maximum of 10 characters long. The file is created in the file directory */opt/SMAW/SMAWcmx/lib/ccp/diagfiles* If a file with this name already exists, it will be overwritten.

  file
  : File name. The file name can be a maximum of 10 characters long. The extension *.bin* is appended within the system.

  If *-f␣file* is not specified, the file is given the default name *cc_DU.bin*, for example, *W2_DU.bin* for a dump from CC *W2*.

You can execute the *Request memory dump...* operation under the *Operations for CC* option (select the CC you want) in the *CCs - Communication Controller* CMX menu, instead of this command.

6.2.2 format - convert trace lists and dumps

The *format* command is used to convert both the trace lists and dumps from binary files into readable form.

Trace list conversion

The trace lists must have been created with *son*, *ton* and *tof*.

Dump conversion

The CC memory must first be dumped into a file (see *dump* command).

format[␣**-b**␣cc]␣[**-c**␣ccp]␣**-t**␣list[␣**-f**␣file1][␣**-g**␣file2]

-b␣cc
: Specification of the CC ID.

-c␣ccp
: Name of the network access software from which the trace list or dump was created.

  ccp
  : Name of the network access software: isdns0 or isdns2

-t␣list
: Specification of the trace list or dump. If you want to enter more than one trace list, enclose the specification in double quotes and separate each list ID with a blank, e.g. -t “IS IN”.

  list
  : ID of the trace list or dump according to the following table.

ID	**Meaning**
IN	NPI trace (profile, TSP, addresses)
IP	X.25 trace, ISDN B-channel
IS	ISDN
LI	Interface signals of layer 2
LP	HDLC
SN	Multilink trace
LM	Layer manager of layer 2

Table 13: Trace list and dump IDs with the format command

ID	Meaning
AL	ADM list of the LMDE
HP	Bus interface controller / host
ER	Error message list
SX	X.25 statistics
LA	Line statistics
LV	Resource statistics
SL	Sorted list
DU	Dump

Table 13: Trace list and dump IDs with the format command

-f␣file1

Specification of the file that contains the dump or trace list.

If -*f*␣*file1* is specified, only one ID may be specified for the -*t*␣*list* option.

file1

Name of the file into which the trace list or dump was written (see the *file* option argument for the *dump* or *ton* command). Only one file with the name *file1_0.bin* and/or *file1_1.bin* may be present within the system for trace list conversion. The name *file1* may be a maximum of 8 characters long.

-f␣file1 not specified:

The default file names are listed in the following table:

Lists	Default file names
Tracelisten	cc_list_0.bin cc_list_1.bin
Dumplisten	cc_DU.bin cc_DU.lay (layer conversion) cc_DU.mem (memory conversion) cc_DU.tsk (task conversion)

Table 14: Default file names with the format command

-g␣file2

Specification of the file into which the converted trace list or dump is to be written.

file2

Name of the file. The file name may be a maximum of 8 characters long. The extension *.txt* is appended to the name within the system.

-g␣file2 not specified:

The file name is created from the name of the file *file1* to be converted by, replacing the extension *.bin* with *.txt*.

Example

You want to create and convert a trace list of layer 2 (LP list) for CC *W1*.

Before starting the test

1. Switch the LP trace list on:

```
W1* son -t LP
```

2. Then switch the list transfer for the LP trace list on:

```
W1* ton -t LP
```

After ending the test

1. Switch the LP trace list off:

```
W1* sof -t LP
```

2. Transfer the sublist that is not completely filled, by switching the list transfer off:

```
W1* tof -t LP
```

3. Convert the transferred trace list:

```
W1* format -c ISDNSO -t LP
```

The result is written into the file: */opt/SMAW/SMAWcmx/lib/ccp/diagfiles/W1_LP.txt*.

6.2.3 sof - switch trace off

You can switch trace lists and trace points off with the *sof* command.

sof[␣**-b**␣cc]␣**-t**␣list[␣**-i**␣id]

-b␣cc

Specification of the CC ID.

-t␣list

Specification of the trace list(s). The individual lists must be separated by blanks and the complete specification enclosed in double quotes, e.g. -t “IS IN”.

list

ID of the trace list according to the following table.

ID	**Meaning**
IN	NPI trace (profile, TSP, addresses)
IP	X.25 trace, ISDN B-channel
IS	ISDN
LI	Interface signals of layer 2
LP	HDLC
SN	Multilink trace
LM	Layer manager of layer 2
AL	ADM list of the LMDE
HP	Bus interface controller / host

Table 15: Trace list and dump IDs with the format command

-i␣id

Specification of the trace list to be switched off, depending on the trace list. If you want to specify several trace points, enclose the specification in double quotes and separate each point ID with blanks, e.g. -i “IS LP”.

id

ID of the trace points according to the following table:

With reference to the IS trace list: a trace can only be switched on in the IS list if a switched connection is in operation. The *sof* command for the IS list is rejected during operation of an ISDN permanent connection.

Trace list	ID	Trace points/meaning
IS trace list Default: SA + PM +SV	ST SV SA SI PM	Splitting overall trace Splitting connection trace Interface trace of the signaling automaton Interface trace of the protocol engine to the signaling automaton Trace of the D-channel protocol elements
LI trace list Default: LC + LB + LY	LC LB LY	ILINK-P trace ILINK-E trace IPHYS line trace
LP trace list Default: LH + LV + LD	LH LV LD LX	HDLC procedure trace Connection handler procedure trace Data trace Extended data trace
AL trace list Default: ID + IM	ID IM	ILMDE trace ILME trace
HP trace list Default: AD + DI + DO	AD DI DO	Administration interface trace ITRANS-IN data trace ITRANS-OUT data trace

Table 16: Trace point IDs with the sof command

-i␣id not specified

The trace lists specified with *list* are switched off.

6.2.4 son - switch trace on

You can switch trace lists and trace points on with the *son* command.

son[␣**-b**␣cc]␣**-t**␣list[␣**-i**␣id]

-b␣cc

Specification of the CC ID.

-t␣list

Specification of the trace list(s). The specification must be enclosed in double quotes, with the lists separated by blanks, e.g. -t “IS IN”.

list

IDs of the trace lists according to the following table.

A trace can only be switched on for the IS list if a switched connection is in operation. The *son* command for the IS list is refused during operation of an ISDN permanent connection.

ID	**Meaning**
IN	NPI trace (profile, TSP, addresses)
IP	X.25 trace, ISDN B-channel
IS	ISDN
LI	Interface signals of layer 2
LP	HDLC
SN	Multilink trace
LM	Layer manager of layer 2
AL	ADM list of the LMDE
HP	Bus interface controller / host

Table 17: Trace list IDs with the son command

-i␣id

Specification of the trace points to be switched on, depending on the trace list. If you want to specify several trace points, enclose the specification in double quotes and separate each point ID with blanks, e.g. -i “SI PM”.

id

ID of the trace points according to the following table.

Trace list	**ID**	**Trace points/meaning**
IS trace list Default: SA + PM +SV	ST SV SA SI PM	Splitting overall trace Splitting connection trace Interface trace of the signaling automaton Interface trace of the protocol engine to the signaling automaton Trace of the D-channel protocol elements
LI trace list Default: LC + LB + LY	LC LB LY	ILINK-P trace ILINK-E trace IPHYS line trace
LP trace list Default: LH + LV + LD	LH LV LD LX	HDLC procedure trace Connection handler procedure trace Data trace Extended data trace
AL trace list Default: ID + IM	ID IM	ILMDE trace ILME trace
HP trace list Default: AD + DI + DO	AD DI DO	Administration interface trace ITRANS-IN data trace ITRANS-OUT data trace

Table 18: Trace point IDs with the son command

-i␣id not specified

The default trace points are switched on for the trace lists specified with *list*.

6.2.5 tof - switch trace list transfer off

The *tof* command is used to switch off the trace list transfer from the CC into trace files. After the *tof* command is input, partly filled sublists from the CC are transferred into the trace files. The transfer is subsequently switched off. The trace lists can be converted with the *format* command.

tof[␣**-b**␣cc]␣**-t**␣list

-b␣cc
Specification of the CC ID.

-t␣list
Specification of the trace list(s). The specification must be enclosed in double quotes and the individual lists separated by blanks, e.g. -t “IS IN”.

list
IDs of the trace lists according to the following table.

ID	**Meaning**
IN	NPI trace (profile, TSP, addresses)
IP	X.25 trace, ISDN B-channel
IS	ISDN
LI	Interface signals of layer 2
LP	HDLC
SN	Multilink trace
LM	Layer manager of layer 2
AL	ADM list of the LMDE
HP	Bus interface controller / host
ER	Error message list
SX	X.25 statistics
LA	Line statistics
LV	Resource statistics
SL	Sorted list

Table 19: Trace list IDs with the tof command

6.2.6 ton - switch trace list transfer on

The *ton* command is used to switch on the trace lists transfer from the CC into trace files. There are two files in each case, which are written to cyclically and alternately (see *file* option).

ton[␣**-b**␣cc]␣**-t**␣list[␣**-z**␣time][␣**-f**␣file][␣**-l**␣length]

-b␣cc

Specification of the CC ID.

-t␣list

Specification of the trace list(s) for which transfer is to be switched on. The specification must be enclosed in double quotes, and the individual lists separated by blanks, e.g. -t “IS IN”.

list

IDs of the trace lists according to the following table.

ID	**Meaning**
IN	NPI trace (profile, TSP, addresses)
IP	X.25 trace, ISDN B-channel
IS	ISDN
LI	Interface signals of layer 2
LP	HDLC
SN	Multilink trace
LM	Layer manager of layer 2
AL	ADM list of the LMDE
HP	Bus interface controller / host
ER	Error message list
SX	X.25 statistics
LA	Line statistics
LV	Resource statistics
SL	Sorted list

Table 20: Trace list IDs with the ton command

-z␣time

Specification of the time period for list transfer.

time

Time period for list transfer, 1..64800 seconds. The list is transferred automatically after this time period.

-z␣time not specified:

Sublist is transferred as soon as it is full.

-f␣file

Specification of the file into which the requested list is to be written. The file is created or overwritten in the *opt/SMAW/SMAWcmx/lib/ccp/diagfiles* file directory.

file

Name of the file. The file name may be a maximum of 8 characters long. Only one trace list can be specified.

There are two files in each case, which are written to cyclically and alternately. The extension *_nr.bin* is created within the system, whereby *nr* (*nr* = 0, 1) identifies both trace files that are written into cyclically.

-f␣file not specified:

The trace list is transferred into files with the default names *cc_list_0.bin* and *cc_list_1.bin*. The files are created or overwritten in the */opt/SMAW/SMAWcmx/lib/ccp/diagfiles* file directory.

-l␣length

Length of the trace file (in bytes), default: 50 000.

6.3 Help functions

The help functions simplify the handling of command files.

6.3.1 cmdfile - execute command file

You can execute a command file with the *cmdfile* command.

cmdfile[␣**-b**␣cc]␣**-f**␣path

-b␣cc
: Specification of the CC ID.

-f␣path
: Specification of the command file.

  path
  : Path name of the command file that contains the command lines. A command line can contain a maximum of 240 printable characters.

Example

Command:

```
cmdfile –f /usr/admin/lplist
```

A command file could appear as follows:

```
# switch default trace points of LP trace list on:
son –t LP
# switch list transfer on for LP trace list:
ton –t LP
```

6.3.2 : Change administered CC

The *:* (colon) help function is used to change the administered CC.

:␣**-b**␣cc

-b␣cc
: Specification of the CC ID that you want to administer.

6.3.3 ? List expert mode commands

The *?* (question mark) help function provides on-screen information for commands that are available in expert mode.

?[␣**-f**␣fct]

-f␣fct
Specification of the function.

fct
Name of the function.

Output of the syntax of the specified command, with all switches and operands.

-f␣fct not specified:
All commands are listed.

6.3.4 ! execute shell command

The *!* (exclamation mark) help function, followed by commands, is used to execute shell commands.

!␣comm

comm
Specification of a shell command line.

Example 1

You are in diagnostic mode of CC *W2* and want to view the *NEWSFILE* diagnostic file.

```
W2* ! cat /opt/SMAW/SMAWcmx/lib/ccp/diagfiles/W2_NEWSFILE_1
```

You are returned to diagnostic mode after the diagnostic file has been output.

Example 2

You are in diagnostic mode of CC *W1* and want to change into a subshell.

```
W1* ! sh
```

6.3.5 # Comment command file

The #␣ (number symbol followed by blanks) help function is used to comment a command file. Lines in the file that begin with this character are recognized as comments and ignored.

#␣[comment]

comment
: Any character string.

7 Administration and diagnostic commands for the Transport Service Providers (TSPs)

This chapter gives information on the administration of the Transport Service Providers (TSPs). The Transport Service Providers use transport protocols to make network transport services available. The product, CCP-OSI/NEA, makes the NEA and TP02 TSPs available. The NTP TSP is a component of CMX.

This chapter provides instructions for diagnosing errors and also instructions on what to do if you want to customize your configuration. You will find full descriptions of the commands and files mentioned in this section in the man pages.

Commands for the TSPs are entered at the operating system level.

7.1 Operational readiness of the Transport Service Providers

Normally, when CMX or CCP/OSI-NEA have been installed and the system is started, the installed TSPs will start automatically. When started successfully, the TSPs are **ready for operation**. The TSPs will then enter the READY state, i.e. they can now be used for data communication. In the following example, the NEA-TSP is ready for operation:

```
#nea
Control Command NEA TSP (SMAWnea 6.0A00) Tue May 13 13:56:17
CEST 2003

    State of the NEA TSP:  READY.
    NEA daemon nead:       running,  PID=4544.
    NEA TSP autostart is:  enabled.
    Periodical check
    of the NEA daemon is:  enabled.
    Local Nsap (proc/reg)  2/18
```

A TSP may fail to start if the resources or required configuration information for starting are not available. For example, the NEA-TSP fetches its local NSAP from the current configuration database. If no local NEA-NSAP is configured, NEA-TSP will not be ready for operation.

A TSP can be interrogated for its current state using the command, *cmxinfo* and the individual commands for each TSP, *nea*, *ntp* and *tp02*.

The state of a TSP that is installed but not ready will be shown as EXIST:

```
#nea
Control Command NEA TSP (SMAWnea 6.0A00) Tue May 13 14:02:32
CEST 2003

    State of the NEA TSP:  EXIST.
    NEA daemon nead:       not running.
    NEA TSP autostart is:  enabled.
    Periodical check
    of the NEA daemon is:  disabled.
```

Additional diagnostic information can be obtained by using the *diag* variant of the TSP commands *nea*, *ntp* and *tp02*. The last entry in the resulting output indicates the configuration error as follows:

```
Nov 23 14:00:56: Start of NEA TSP failed: No local NSAP
configured.
```

The TSPs can be started and stopped manually. It should be noted that when a TSP is stopped, all the active connections using it will be lost. These actions are available in the CMXCUI menu, *Transport Service Provider*.

Command syntax:

nea [**diag** | **start** | **stop**]

ntp [**diag** | **start** | **stop**]

tp02 [**diag** | **start** | **stop**]

start

starts the TSP after various checks. A message will be sent to default output if this TSP is already loaded.

start also starts those entries in the *crontab* file which in turn start the actions which are to be carried out a regular intervals. These check whether the TSP is still running and if not, restart it. These actions continue to be carried out until the TSP is stopped explicitly by the *stop* command or until three successive failures to restart have occurred. In both cases, the automatic execution of these actions will be stopped and an error entry will be written in the log file.

stop

stops the TSP. The operation outputs a message if the corresponding TSP is not loaded. If a TSP is stopped, it releases its reserved system resources. In this state, no communication is possible using this TSP. Regular actions called from the system's *crontab* file are also stopped. This status remains unchanged until the next start.

diag

Outputs a log file for the corresponding component.

Files:

Replace the following file name, *$Name*, by the name of the corresponding TSP: NEA, NTP or TP02.

/var/opt/SMAWcmx/adm/log/$Name.log

log file of the corresponding TSPs.

7.2 TSP statistics

Each TSP maintains an internal count of the number of connections, data transferred etc. This information can be called using the commands *neastat*, *ntpstat* or *tp02stat*. If the commands are given with the *-r* option, the counters are reset to their starting value, *0*.

Command syntax:

neastat [**-r**]

ntpstat [**-r**]

tp02stat [**-r**]

-r

Rests the counter values.

7.3 Switching NEA routing on and off

You can switch NEA routing on or off or request the current status using the *neatune routing* command. The NEA routing becomes active immediately when it is switched on. However, deactivation using *neatune routing = off* only takes effect when the system is restarted. The setting are stored even after system shutdown.

neatune routing[= **on**| **off**]

neatune routing
If no option is selected the current status is shown (routing on or off).

neatune routing = on
NEA routing is switched on.

neatune routing = off
NEA routing is switched off.

8 Diagnostics and Fault-finding

When CCP-ISDN runs or a CC is loaded, status information is automatically collected and written to various lists or files. The information contained in them is a valuable aid to fault-finding and system monitoring. The following information is logged:

– Traces

  The traces log the CCP-ISDN process steps. The traces are recorded in layer-specific lists, which can be explicitly activated and can include a wide range of trace points.

– Statistics

  Statistical lists provide information on resource loading and failure rates, for example for the line and memory area. Operational process statistical lists, for example, can yield a great deal of information on an ISDN connection with an X.25 network.

– Diagnostic files

  Diagnostic files are continuously updated with error messages, records of access protection violations and other important messages regarding the subnet profile.

The following diagnostic and fault-finding tools are also available:

Tool	**Brief description**
x25snoop	Switches on the line-specific X.25 protocol traces and off-line processing of the trace, using the Ethereal protocol analyser (available as freeware)
ccptron, ccptroff	Switch the most important CC traces on and off

Table 21: Diagnostic and fault-finding tools

8.1 Diagnostic files

The relevant diagnostics files for the CC and the subnet profile are described below. The files do not require to be explicitly activated or deactivated. With the exception of the ER list, these files can be read without further processing.

The error message field (ER list)

The occurrence of serious CC errors and warnings is logged in the error message field (ER list). The list transfer for the ER list is switched on by default, i.e. you do not need to enter the *ton* command.

You must use the *tof* command to request a partial (incomplete) list. You can use the *format* command to process the error message list to a readable form. As default, the trace lists are stored under *opt/SMAW/SMAWcmx/lib/ccp/diagfiles/cc_ER_nr.bin*, where *nr* = 0.1.

If, after entering the *tof* command, further ER lists are to be transferred, you must first enter the *ton* command.

The file, NEWSFILE

Messages relating to the subnet profile are logged in the *NEWSFILE* file. There are two files that are written in parallel. They are stored under the name, *cc_NEWSFILE_nr*, in the directory, *opt/SMAW/SMAWcmx/lib/ccp/diagfiles*, where *nr* = 0.1.

The messages appear as follows:

```
4108 NEWS FROM PROCESSOR 2/142: SYSTEM READY!
4270 NEWS FROM PROCESSOR 2/142: PROCESSOR LINE ACTIVE,LINE 0
```

This message shows which of the generated lines are really active and which are not. If a line is not active, there is no point in attempting to establish a connection over it.

The file, DEBUGFILE

The file, *DEBUGFILE* contains information on the subnet profile and on the start time.

Access attempts by unauthorized partners are also recorded in the DEBUGFILE. Other information recorded includes date, time and addresses. Note: the DEBUGFILE is deleted when the CC is reloaded.

There are two files that are written in parallel. They are stored under the name, *opt/SMAW/SMAWcmx/lib/ccp/diagfiles/cc_DEBUGFILE_nr*, where nr = 0.1.

8.2 Traces and statistics

Traces

If you wish to generate trace lists, you must first enter the *son* and *ton* commands for the desired lists. With the *son* command, you can switch on trace lists and with the *ton* command, you activate the trace list transfer from the CC to a UNIX file. The transfer of the trace lists takes place at fixed intervals and always provided a list is full.

After testing, enter the commands *sof* and *tof*, where the *sof* command turns off the trace for a specific list and the *tof* command transfers the still incomplete lists from the CC. The transfer will then be switched off.

By default, the trace lists are stored under the name, *cc_list_nr.bin*, where *nr* = 0.1. The trace information is not printable but can be prepared for printing with the *format* command.

If, after entering the commands *sof* or *tof*, further trace lists are to be generated, you must enter the *son* and *ton* commands again.

The following tables summarise the traces and statistics and their meanings.

Identification	Meaning
ER	Error message list **)
IN	NPI information on profile, TSP and addresses
IP	X.25 trace, ISDN B-channel
IS	ISDN
LP	HDLC
SL	sorted list *)
DU	dump *)

Table 22: Important traces

Identification	Meaning
SN	Multilink trace
LM	Layer Manager of layer 2
LI	Interface signals of layer 2
LN	Line component control signals
HP	Controller / host bus interface
AL	ADM list of the LMDE

Table 23: Other traces

The trace and statistics IDs can be given for the commands *format*, *son*, *sof*, *ton* and *tof*.

*) only for the *format* command.

**) for the *son* and *ton* commands, these IDs are already set automatically when the CCP starts

Statistical lists

Statistical lists are like trace lists in that you must switch on the transfer with the *ton* command and reset the value of the statistical lists to 0 at the start of the transfer.

If you enter the *tof* command, the most recent list entries (residual list) are transferred and the transfer is then switched off.

The lists are, for example, stored under *W2_LA_0.bin* or *W1_LV_1.bin*.

Identification	Meaning
SX	X.25 ***)
LA	line ***)
LV	resources ***)

Table 24: Statistics

The trace and statistics IDs can be given for the commands *format*, *son*, *sof*, *ton* and *tof*.

***) for the *son* command, these IDs are already set automatically when the CCP starts.

8.3 ccptron, ccptroff – Switching traces on and off

The *ccptron* command, switches on the most important traces of a CC. *ccptron* automatically determines which are the traces to be switched on. In the case of a WAN controller, these are the traces IN, IP and LP. In the case of an ISDN controller, the traces are IN, IP, IS and LP. If no arguments are given, *ccptron* switches on the traces for every CC that, according to the *bstv info* command, is in the READY state.

ccptroff is used to switch the traces off again and convert them into a readable format. The *ccptroff* command also prepares the error message field (ER list).

The readable and binary trace files are written to the directory, */opt/SMAW/SMAWcmx/lib/ccp/diagfiles*.

ccptron [␣**-b**␣cc]

ccptroff [␣**-b**␣cc]

-b cc
defines a CC in the form, Wn, n = 1,...

Example:

```
ccptron -b W13
Attention: Previous trace lists are removed
ccptron/W13 : son && ton started for trace lists "in ip lp".

ccptroff -b W13

Analysis of TRACE list  IN  into file:
/opt/SMAW/SMAWcmx/lib/ccp/diagfiles/W13_IN.txt

Analysis of TRACE list  IP  into file:
/opt/SMAW/SMAWcmx/lib/ccp/diagfiles/W13_IP.txt

Analysis of TRACE list  LP  into file:
/opt/SMAW/SMAWcmx/lib/ccp/diagfiles/W13_LP.txt

Analysis of TRACE list  ER  into file:
/opt/SMAW/SMAWcmx/lib/ccp/diagfiles/W13_ER.txt
```

8.4 x25snoop – Carrying out line-specific X.25 protocol tracing

The *x25snoop* command provides diagnostic assistance for errors in connection with X.25. It provides line-specific tracing on the controller and off-line preparation of the protocol elements sent to the partner or received from the partner over this line. The output is prepared in user-readable form by using the Ethereal freeware graphic network protocol analyser. In addition to HDLC and X.25, the TP02 and TCP/IP logs are prepared. *x25snoop* can be used for:

– X.25 connections

– DTE-DTE couplings

– X.31

– X.32 2 level selection

– X.25 fixed connection to an ISDN controller

Given that *x25snoop* records the protocol elements exchanged over a line, *x25snoop* is not suitable for the analysis of local connection refusals.

x25snoop -s

Switches on a trace (LP data trace: bstv son -b<cc> -tlp -ilx -l<line>) on the controller, that copies the first 264 Bytes of each HDLC frame directly from the line. X.25 packets up to 256 Bytes in length are included in full. For longer packets, the first 259 Bytes of each X.25 data packet are included. The trace data are written periodically in two binary trace files whose size you can set with the *-m* option.

Do **not** switch on the LP trace on the controller at the same time. *x25snoop* cannot be used for several lines of a controller simultaneously.

x25snoop -t

Stops the trace and generates a file from the two binary trace files that can be prepared using Ethereal. This is saved in the standard directory for the controller traces and is named:

/opt/SMAW/SMAWcmx/lib/ccp/diagfiles/<cc>_IF<line>.rad <cc> = W1,...

In the case of an X.25 via ISDN where there are a number of B-channels, *x25snoop* generates an Ethereal input file for each B-channel on which there has been data traffic. The signals on the D-channel are not traced.

/opt/SMAW/SMAWcmx/lib/ccp/diagfiles/<cc>_IF<line>_B1.rad

/opt/SMAW/SMAWcmx/lib/ccp/diagfiles/<cc>_IF<line>_B2.rad

x25snoop -s -i

Instead of switching on the trace with *x25snoop -s* and switching it off with *x25snoop -t* , you can enter *x25snoop -s -i*.

In this case, press the DEL key, to turn off the trace.

Switch trace on	**Switch trace off**
x25snoop -s [-b *cc*] [-l *line*]	x25snoop -t [-b *cc*]
or	
x25snoop -s -i [-b *cc*] [-l *line*]	DEL

Trace file processing with Ethereal:

Ethereal is a network protocol analysis tool which you use to turn the trace data generated by *x25snoop* into a user-friendly form. You can either use *ethereal* to display the trace in a graphic window or *tethereal* to present it in text form.

For the method of calling these programs and the appearance of their outputs, see the examples on page 168 and page 171. There you will also find some notes on the graphic display with *ethereal*.

Ethereal is available as freeware for Solaris, Windows and some other operating systems (www.ethereal.com). For Solaris, Ethereal is shipped on the installation CD for CMX. Install the following packages:

– SMAWPbase

– SMAWPglib

– SMAWPgtk+

– SMAWPethe

With the standard installation of CMX, these packages are automatically installed. There are also man pages for *ethereal* and *tethereal* that can be called with *man*, as well as documentation in HTML, located in the directory, */opt/SMAWPlus/readme/SMAWPethe/html*.

Command syntax:

x25snoop -s [␣**-i**] [␣**-b**␣cc] [␣**-l**␣line] [␣**-m**␣size]

x25snoop -t [␣**-b**␣cc]

-s

Starts the trace for the selected line. The trace data are written periodically in two files whose size you can set with the *-m* option.

Example for cc *= W1:*

/opt/SMAW/SMAWcmx/lib/ccp/diagfiles/W1_LP_0.bin

/opt/SMAW/SMAWcmx/lib/ccp/diagfiles/W1_LP_1.bin

-i

x25snoop does not stop after the trace has been switched on but continues to output the size of the binary trace files every second. If you press DEL, the trace will be switched off and converted to the RADCOM format (see *-t* option).

-t

Stops the trace and generates an input file for Ethereal. The file is written to the standard directory for the controller traces. The file name includes the CC and the line.

Example for cc *= W1:*

/opt/SMAW/SMAWcmx/lib/ccp/diagfiles/W1_IF<line>.rad

In the case of X.25 via ISDN, *x25snoop* generates B-channel-specific files.

Example for cc *= W2 and* line *=1:*

/opt/SMAW/SMAWcmx/lib/ccp/diagfiles/W2_IF1_B1.rad

/opt/SMAW/SMAWcmx/lib/ccp/diagfiles/W2_IF1_B2.rad

In the case of a fixed X.25 connection to a PWS0 controller, the forward slash in the line number is left out when forming the file name:

Example

/opt/SMAW/SMAWcmx/lib/ccp/diagfiles/W3_IF10.rad

-b␣cc

defines a CC in the form, Wn, n = 1,...

If nothing is entered, the CC is requested. The request includes the valid values.

-l␣line

Line number. The line number to be given can be found using *bstv linkstat -b␣cc* under IF#. Possible values are:

PWXV-2: line = 1, 2

PWXV-4: line = 1, 2, 3, 4

PWS0: line = 1, 2, 1/0, 1/1, 1/2, 2/0, 2/1, 2/2

PWS2: line = 1

If nothing is entered, the line number is requested. The request includes the valid values.

-m␣size

Maximum size of the binary trace files in Bytes.

Default value: 50,000 Bytes per file

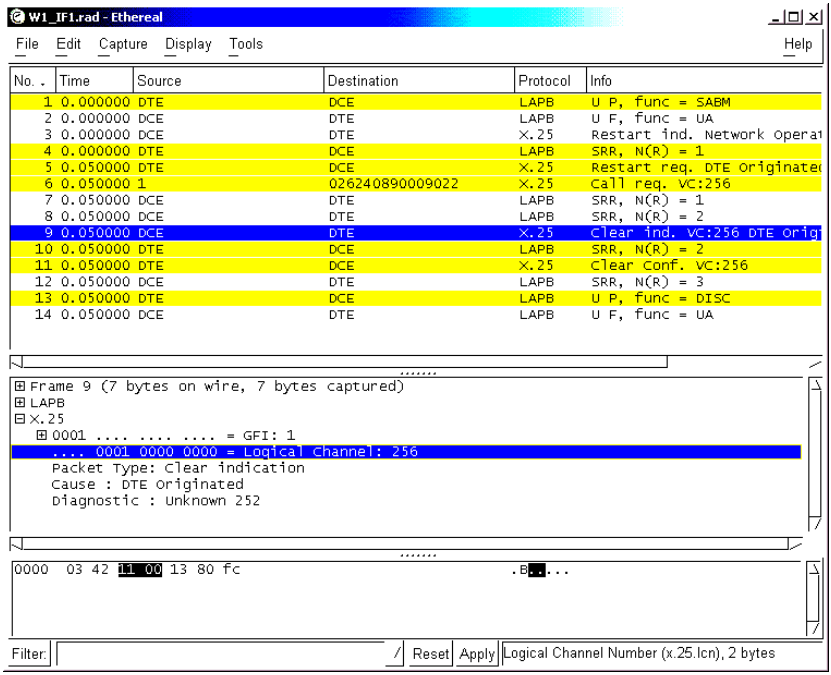
Example:

```
x25snoop -s -i
Enter CC ( W1 | W2 ): W1
Enter line number ( 1 | 2 ): 1
Starting the LP trace on W1 line 1.
Trace successfully started.

Interactive mode:
x25snoop shows the sizes of the binary trace files and
updates
the values every second. Type DEL when the test is finished
and
you want to stop the tracing.

File names:
/opt/SMAW/SMAWcmx/lib/ccp/diagfiles/W1_LP_0.bin
/opt/SMAW/SMAWcmx/lib/ccp/diagfiles/W1_LP_1.bin

File sizes:   400  /  0

DEL

Starting the LP trace on W1 line 1.
File sizes:   608  /  0

TRACE list LP converted, file:
/opt/SMAW/SMAWcmx/lib/ccp/diagfiles/W1_IF1.rad
```

Example for the graphical display with ethereal in Solaris:

```
DISPLAY=<Name des X-Servers>:0
export DISPLAY
ethereal -r /opt/SMAW/SMAWcmx/lib/ccp/diagfiles/W1_IF1.rad
```

The example shows an outgoing connection request (Call Request), that is refused (Clear Indication). The Clear Indication packet is marked.
The bottom window shows the marked Clear Indication packet, including the LAPB header, in hexadecimal form. The middle window shows the detailed X.25 protocol structure of the packet.



Setting colour markings (e.g. for transmit and receive directions):

You can define a colour marking for anything that can be filtered.

Colour-mark transmitted frames:

- In order to set colour marking, select the *Display* menu -> *Colorize Display*.
- Select *New* and enter a name.
- In order to fill in the *String* field, delete its contents and select *Add Expression*.
- Open *Frame*, *Point-to-Point Direction*, ==, *Sent* and *Accept*.
- Select *Background Color*, a colour and then click twice on *OK*.
- Click on *Apply* and then on *OK*.
- Select *Save*, to store this setting for the next call of *ethereal*.

Activating filtering (e.g. by protocol or VC):

Use one of the following methods:

- To activate a simple filter, write the desired filter string in the *Filter* field at bottom left and click on *Apply*.

  Examples:

x.25	for X.25 protocol
cotp	for TP02 protocol
x.25.lcn == 10	for all X.25 packets on VC 10

or, alternatively:

- Mark a line in the middle window.

  An expression will appear in the grey field at the bottom right.

- Use this expression as the filter.

- In order to activate this filter, select the *Display* menu -> *Match* -> *Selected*.

Defining a filter, using it, saving it for the next call and deleting it:

- Click in the filter field *Filter* at bottom left.

- To define a new filter, fill in the fields, *Filter name* and *Filter string* – using the option *Add Expression* – and click on *New*.

  The initial value of the *string* filter field is the currently active filter.

Show time:

The default setting of the time display is *Seconds since beginning of capture*, i.e. the time shown is the time from the first frame.

- To change the setting of the time display, open the menu *Display* -> *Options* and select one of the options *Time of day* or *Date and time of day*.

Example of a trace displayed as text, using tethereal:

```
tethereal -r /opt/SMAW/SMAWcmx/lib/ccp/diagfiles/W1_IF1.rad

 1  0.000000       DTE -> DCE          LAPB U P, func = SABM
 2  0.000000       DCE -> DTE          LAPB U F, func = UA
 3  0.000000      DCE -> DTE         X.25 Restart ind. Network
Operational - Diag.:7
 4  0.000000      DTE -> DCE         LAPB SRR, N(R) = 1
 5  0.000000      DTE -> DCE         X.25 Restart req. DTE
Originated - Diag.:0
 6  0.000000        1 -> 026240890009022 X.25 Call req. VC:10
 7  0.000000      DCE -> DTE         LAPB SRR, N(R) = 1
 8  0.000000      DCE -> DTE         LAPB SRR, N(R) = 2
 9  0.000000      DCE -> DTE         X.25 Clear ind. VC:10 DTE
Originated - Unknown 252
10  0.000000      DTE -> DCE         LAPB SRR, N(R) = 2
11  0.050000      DTE -> DCE         X.25 Clear Conf. VC:10
12  0.050000      DCE -> DTE         LAPB SRR, N(R) = 3
13  0.050000      DTE -> DCE         LAPB U P, func = DISC
14  0.050000      DCE -> DTE         LAPB U F, func = UA
```

Glossary

1TR6

Technical standard number 6 (National ISDN Telekom Germany)

API (Application Program Interface)

APIs are program interfaces that provide the functions of a program system. A programmer uses the APIs when programming applications. APIs provide functions for connection management, data interchange and for representation of names within addresses. Sockets, ICMX and XTI are APIs in the CMX environment.

Applications

An application is a system of programs that uses specific services of an EDP system in order to provide a human or machine user with a higher level service. Communication applications are applications that use the communication functions of an EDP system together with a network to provide system-transcending services.

A prefix is added to most applications to define the underlying service (*CMX application*, UTM application, DCAM application, Motif application, Windows application, etc.). Examples of communication applications are file transfer, terminal emulation, electronic mail, World Wide Web browsers and servers, transaction systems such as UTM, in general all applications that work according to the client/server principle.

See also *TS application*.

ASCII-Code

International 7-bit based character set for DP systems (ISO 7-bit code).

CC (Communications Controller)

A CC is a module for connecting a UNIX computer to a network. You require a CC to physically connect your computer to a subnet, unless the connection is integrated into another module, e.g. the motherboard (onboard connection). CCs are generally operated together with a corresponding *Communication Control Program* (*CCP*) to obtain a logical connection to the network. These CCs are known as loadable CCs. Loadable CCs are generally controlled by a *subnet profile*, which is an integral part of the *CCP*.

CCP (Communication Control Program)

A CCP is a program system (software product) which, together with one or more *CCs*, provides a UNIX computer with logical access to a *network*. A CCP implements the four lower layers of the OSI Reference Model (transport system) for data communications. CCP-WAN and CCP-ISDN are examples of CCPs for connection to X.25 and telephone networks, and ISDN.

A CCP consists of a number of components, the *subnet profile* and the *Transport Service Providers*.

CCP-ISDN

Generically used designation for the CMX/CCP products used to implement ISDN communication: CMX, CCP-ISDN-LINK and CCP-OSI/NEA (optional).

CCP profile

A CCP profile defines a specific protocol for each of the four lower layers in the *ISO Reference Model*, thereby defining specific network characteristics. A CCP contains at least one CCP profile.

CCP-WAN

Generically used designation for the CMX/CCP products used to implement WAN communication: CMX, CCP-WAN-LINK and CCP-OSI/NEA (optional).

CLI (Command Line Interface)

CLI is the sum of commands for *OA&M* of *CMX* and the *CCPs*. As the administrator, you can execute initialization, monitoring, control and maintenance functions of *CMX*, the *CCPs* and the *communication services* via the UNIX command line (the commands cmxinfo, cmxm(onitor), tnsxcom, bstv, ccpgen, etc.).
CLIs offer a wide spectrum of options, some with complex syntax. The user interface *CMXCUI* enables simple, interactive handling of the required routine tasks.

CMX (Communication Manager UNIX)

CMX provides communications services for using *CMX applications* and *communication services* in the network and enables programming of CMX applications. CMX unifies the services of different networks, thereby allowing the same CMX application to be used regardless of the underlying network. As a runtime system, CMX mediates between the current network and CMX applications and offers the network administrator unified functions for *OA&M* (Operation, Administration and Maintenance) of *CCPs* and *CCs*. As a development system, CMX provides interfaces (APIs) and processes for programming network-independent CMX applications.

CMX applications

CMX applications are applications that use the CMX services. They have an address in the network, the *TRANSPORT ADDRESS*. They identify themselves to each other with symbolic names, the *GLOBAL NAME* of an application.

CMXCUI (Character User Interface)

The CMXCUI is a character-oriented user interface for the *OA&M* functions of *CMX* and the *CCPs*. As the administrator, you can comfortably operate the *OA&M* via menus and forms. The CMXCUI uses FMLI and interfaces onto the *CLI*.

Communication partner

A *TS application* which maintains a logical connection to another *TS application* and exchanges data with it.

Connection, logical

Assignment of two *communication partners*, enabling them to exchange data.

FSS (Forwarding Support Service)

The FFS is a *CMX* component that supports the correct addressing of applications in the network and the selection of a route through the *network* and its subnets. As the administrator, you can configure the FSS with the network-specific entries that you have foreseen for your network or have agreed with the network provider.

One important piece of information in the FSS is the representation of a network address, e.g. NEA address “47/11” on a subnet address of the remote computer e.g. the X.25 address “8963647658”. A further important piece of information is the definition of a route with its local

starting point and the various stations through the subnets to the remote computer. The local starting point of a route is a subnet ID that uniquely identifies one specific subnet access from several that are available.

GLOBAL NAME of an application

Each CMX application identifies itself and its communication partner in the network by means of symbolic, hierarchic GLOBAL NAMES. A GLOBAL NAME consists of up to five name parts (NP[1-5]) that you can use to define the application (NP5), the computer (NP4) and (up to three) administrative domains (NP[3-1]).
Example: The GLOBAL NAME “YourApplication.D018S065.mch-p.sni.de” means:
“YourApplication” resides in host “D018S065” in domain “mch-p.sni.de”.
As the administrator, you must observe the specifications and recommendations of the special application when selecting a GLOBAL NAME.
As the administrator, you can assign the GLOBAL NAME of an application 1:1 a *TRANSPORT ADDRESS* or *LOCAL NAME* of an application. As a programmer, you can derive the TRANSPORT ADDRESS or LOCAL NAME expected by CMX from the GLOBAL NAME, using the *Transport Name Service* (TNS) function calls.

ISO Reference Model

Model for open systems communication. This is described in the ISO 7498 standard and contains 7 layers.

KOGS (configuration oriented generator language)

KOGS is the configuration oriented generator language with which the physical and logical characteristics of the subnet connections of a computer are described in a text file. The KOGS language elements are macros, operands and operand values. The system or network administrator normally defines the specific characteristics of his subnet interfaces with the *CMXCUI*. He only uses the KOGS for this in exceptional cases.

LOCAL NAME of an application

A CMX application signs on to CMX for communication in your local computer with the LOCAL NAME. The LOCAL NAME consists of one or more T-selectors that each designate the transport system over which the CMX application is to communicate. As the administrator, you can use the LOCAL NAME to enable or disable the communication of a CMX application over specific transport systems and fulfill any requirements of the CMX application for specific T-selector values, e.g for file transfer.

Example: an application is to use the T-selector “cmxappl” (lower case) for communication via the TCP/IP-RFC1006 transport system and the T-selector “$CMXAPPL” (upper case) for communication via the NEA transport system.
As the administrator in CMX, you can assign the LOCAL NAME of an application to the *GLOBAL NAME* of the application. As a programmer, you can derive the LOCAL NAME expected by CMX from the GLOBAL NAME, using the *Transport Name Service* (TNS) function calls.

Message

A logically related data set which is to be sent to a *communication partner*.

Network

A network is a linkage of interrelated transfer components (lines, exchange nodes, processes) with uniformly defined services, protocols and access setups for EDP systems. A network connects computers with one another so that they can use system-independent applications. The network of a network provider can be used immediately for applications or for defining private network structures that build on and overlay it. The following networks are relevant in the UNIX environment: the TCP/IP network, and the SNA, TRANSDATA and OSI networks.
The networks named above as examples can be overlays of public or private subnets such as the X.25, telephone, data, ISDN or ATM networks and various private, local networks based on Ethernet, Token Ring and FDDI.

Network access software

The network access software comprises the components of a *CCP* that are loaded on and control a *Communication Controller*.

Network address

Each computer in a *network* is uniquely identified by its address. A computer can be linked into different networks and then has a specific network address for each of these networks. In the Internet, a computer has an address (IP address) that is made up of the network and host number (e.g. 129.144.89.171). In NEA networks, a computer has an NEA address that is made up of computer/region numbers (e.g. 124/213). The OSI network address (NSAP address) is made up of the Initial Domain Part (IDP) and Domain Specific Part (DSP) and has the format IDP+DSP (e.g. 470058+01444581000073911003080014119613 01).

OA&M (Operation, Administration and Maintenance)

OA&M is the sum of the functions for startup, operation monitoring and control, configuration and maintenance of the CMX and CCP components. The main OA&M activities in the CMX environment are loading and monitoring a CC, configuring the CCP runtime parameters and switching traces.

The simple, interactive handling of routine tasks in OA&M are provided by the *CMXCUI*. You can also use the CLI for special, unusual administration tasks.

OSI Reference model

Open Systems Interconnection is the communication architecture defined by the International Standards Organization ISO in the ISO 7498 standard. It defines reliable data interchange between applications that run on different hardware platforms. The OSI Reference Model defines seven part tasks for solving this complex overall task. These are arranged in hierarchical layers with each of these part tasks being carried out by a specific layer. The lower four layers represent the *transport system*, the upper three layers represent the viewpoint of the *application*, e.g. the data formats.

PDN application

A *TS application* that runs in TRANSDATA PDN on a communications computer.

Point-to-point protocol

Standardized method with which datagrams are routed via TCP/IP (RFCs 1171, 1172 and 1661).

Process

A process is the execution of a program. It consists of the executable program, the program data and process-specific administration data which is necessary to control the program.

Processor

TRANSDATA instance in the host or communication computer which can be addressed from any point in the network and in which the transport services are provided.

Processor name

Part of the *TRANSDATA address*. The processor name has the following syntax:
processor number/region number.

Route

Route describes the path from the local computer to a remote computer within a *subnet*. If the remote computer is in a different subnet from the local computer, the route then describes the path from the local computer to the network interconnection (“next hop”) from which the further routing to the remote computer takes place. A route is defined by its end points: the *subnet ID* of the local computer and the *subnet address* of the remote computer if the remote computer is in the same subnet, or the subnet address of the “next hop” if the remote computer is not in the same subnet.
If a computer has more than one subnet address, it can be accessed via more than one route.

Software configuration

This is a defined combination of versions of software products that together cover a limited and verified performance spectrum.

A software configuration of CMX and CCP product versions guarantee their defined interoperability. This is ensured by quality control measures. Unexpected dropouts and failures with undefined results can occur if CMX and CCP product versions are mixed that are not expressly defined as compatible.

Subnet

A subnet is a technically or administratively homogeneous part of a *network*. Subnets include the X.25, telephone, ISDN and ATM networks and various private, local networks based on Ethernet, Token Ring and FDDI. A subnet can be accessed via one or more subnet interfaces. A subnet interface is identified by its *subnet address*.

Subnet address

The subnet address uniquely describes one subnet access that allows access to the *subnet*. An ISDN subscriber number, a DTE address or an Ethernet address are examples of subnet addresses.

Subnet ID

The subnet ID, also known as SNID, describes a group of similar subnet accesses that lead to the same *subnet*. The subnet ID defines the type of subnet and identifies the group of accesses to this subnet which it comprises. A subnet ID can, for example, stand for two ISDN connections or a number of X.25 connections in one subnet.

T-selector

The T-selector identifies a communications application within the computer on which the application runs. The T-selector, together with the *network address* of the computer, forms the *TRANSPORT ADDRESS* of an application with which the application can be uniquely addressed within a network. The T-selector format and value range depend on the type of *network*. The T-selector corresponds to the station name in the NEA network (e.g. T'DSS01').

TNS (Transport Name Service)

The TNS is a *CMX* component that supports the correct representation of the *GLOBAL NAMES* of *CMX applications* in the network, into *TRANSPORT ADDRESSES* and *LOCAL NAMES*. As the administrator you configure your selected assignment of GLOBAL NAME to TRANSPORT ADDRESS for remote applications and the assignment of GLOBAL NAME to LOCAL NAME for local applications. As an applications programmer you can use these representations via an API and thereby work with the GLOBAL NAME of applications alone, without evaluating the representations. The TNS offers application identification throughout the network by means of logical GLOBAL NAMES and their representation in a corresponding *network address*. This allows you to decouple the applications from knowledge of their network address. The TNS together with the FSS offer complete representation of the logical name in a concrete *subnet address* and *route* through the various subnets in the network.

TNSADMIN

Acquisition program for the *Transport Name Service in UNIX*.

TRANSPORT ADDRESS of an application

A calling *CMX application* passes the TRANSPORT ADDRESS of a called communication partner to *CMX* during connection setup. CMX uses the TRANSPORT ADDRESS to localize the communication partner in the network and define a *route* through the network. The TRANSPORT ADDRESS depends generally on the logical and physical structure of the

network (and its subnets). The TRANSPORT ADDRESS contains the network-specific specifications of its network provider(s). As the administrator, you can influence the TRANSPORT ADDRESS and the communication route, independent of the application.
Integral parts of a TRANSPORT ADDRESS are: a network address for unique identification of the remote computer on which the application resides, the type of the *transport system* over which the remote application can be reached and the *T-selector* which identifies the remote application in the remote computer.
Examples of network addresses are: the Internet address in the point notation “192.11.44.1”, the NEA address in the processor/region number notation”47/11” and the X.25 address (DTE address) as a numeric string “45890010123”.

As the administrator, you can assign the GLOBAL NAME of an application 1:1 a *TRANSPORT ADDRESS*. As a programmer, you can derive the TRANSPORT ADDRESS expected by CMX from the GLOBAL NAME, using the *Transport Name Service* (TNS) function calls.

Transport connection

Assignment of two communication partners that enables them to exchange data.

Transport system

The transport system designates the lower four layers of the *OSI Reference model*. A *CCP* implements the four layers of the transport system. The transport system ensures secure data interchange between computers whose *applications* communicate with each other, independent of the underlying network structures. The transport system uses protocols for this.

TS application

This is an application that uses the services of the transport system. It consists of programs that can set up a logical *connection* to another TS application in order to exchange data with it.

TS directory

Database containing information about the *TS applications*. The TS directory is managed by means of the *Transport Name Service*

TSP (Transport Service Provider)

A TSP is a component of a CCP or CMX that provides the OSI transport services, apart from the NTP (Null Transport), in the network by means of a transport protocol. As the administrator, you can define the use of a specific TSP for the communication of *applications*. RFC1006 is the TSP in CMX that provides the OSI transport services, together with TCP/IP in the Internet. The NTP offers CMX applications direct access to the network services of the X.25 subnet. TP0/2 and NEA are the TSPs for an OSI environment and the TRAANSDATA network.

A TSP together with a *subnet profile* forms a *transport system*. It provides a set of configurable runtime and tuning parameters, evaluates the *TRANSPORT ADDRESS* and finds a suitable route through the network. The TSP uses your entries in the *FSS* if necessary.

Abbreviations

1TR6
signaling protocol for the German national ISDN

ASCII
American Standard Code for Information Interchange

BAC
balanced operation asynchronous balanced mode class

BCAM
Basic Communication Access Method

BTX
videotex (in Germany)

CC
Communications Controller

CCITT
Comite Consultatif International Telegraphique et Telephonique

CCP
Communication Control Program

CF
configuration file

CMX
Communications Manager UNIX

CONS
Connection-oriented Network Service

CUG
Closed User Group

DCAM
Data Communication Access Method

DCE
Data Circuit Terminating Equipment

DSS1
ISDN signaling protocol used in Euro-ISDN amongst others (Digital Subscriber Signalling System No. one)

DTE
Data Terminating Equipment

EAZ
terminal device selection digit (German national ISDN)

EBCDIC
Extended Binary Coded Decimal Interchange Code

ECMA
European Computer Manufacturers Association

EMDS
emulation of data display terminals

FT
file transfer

FV
permanent connection

GSM
Global System for Mobile Communication

HDLC
High Level Data Link Control

IU
Interworking Unit

ISDN
Integrated Services Digital Network

ISO
International Organization for Standardization

ITU
International Telecommunication Union

ITU-T
Telecommunication Standardization Sector

KOGS
configuration orientiert generator language

KR
communication computer

LAPB
Link Access Procedure B

MSN
Multiple Subscriber Number

NEA
network architecture

NEAN
layer 3 protocol

NEATE
layer 4 protocol

OSI
Open System Interconnection

PAD
Packet Assembly Disassembly

PDN
program system for remote data processing and network control

PPP
point-to-point protocol

PVC
Permanent Virtual Circuit

SPV
semipermanent connection

SR
station computer

SVC
Switched Virtual Call

TIDU
Transport Interface Data Unit

TNS
Transport Name Service

TNSADMIN
management program for the Transport Name Service

TPDU
Transport Protocol Data Unit

TPID
Transport Protocol Identifier

TS
Transport System

TSAP
Transport Service Access Point

TSDU
Transport Service Data Unit

TSP
Transport Service Provider

UTM
universal transaction monitor

VAR
host computer

VR
front-end processor

WAN
Wide Area Network

WV
switched connection

XTI
X/Open Transport Interface

Related publications

The manuals are available as online manuals, see *http://manuals.fujitsu-siemens.com*, or in printed form which must be payed and ordered separately at *http://FSC-manualshop.com*.

[1] **CMX V6.0** (Solaris)
Operation and Administration
User Guide

Target group
System administrators

Contents
The manual describes the function of CMX as mediator between applications and the transport system. It contains basic information on configuration and administration of systems in network environments.

[2] **CMX V6.0**
Programming Applications
Programmer Reference Guide

Target group
Programmers

Contents
The manual describes the program interface of CMX, i.e. all tools that you can use for developing TS applications.

[3] **CMX V6.0** (Solaris)
TCP/IP via WAN/ISDN
User Guide

Target group
Network and system administrators.

Contents
The manual describes how CMX enables the connectionless IP traffic via the connection-oriented WAN.

[4] **CMX/CCP V6.0** (Solaris)
WAN Communication
User Guide

Target group
Network administrators and system administrators

Contents
The manual describes the computer-to-computer connection via WAN (Wide Area Network) allowing communication in the remote area (Wide Area Network, WAN).

Other publications

[5] **„CCITT Yellow/Blue/Red Book“**

CCITT Yellow Book

Recommendations of X.3, X.25, X.28, X.29 X.31 and X.32 on packet-switched data transmissions services

UIT Genf 1980

CCITT Red Book

Recommendations of X.3, X.25, X.28, X.29 X.31 and X.32 on packet-switched data transmissions services

UIT Genf 1984

CCITT Blue Book

Recommendations of X.3, X.25, X.28, X.29 X.31 and X.32 on packet-switched data transmissions services

UIT Genf 1988

[6] CCITT-Empfehlungen“
CCITT Empfehlungen der V-Serie und der X-Serie Datenübermittlung (Übersetzungen) Tietz, W.
5. Auflage Band 1 bis Band 7
R. v. Deckers Verlag, G. Schenk
Heidelberg 1985 - 1987

[7] **ISDN-Digitale Netze für Sprach-, Text-, Daten-, Video- und Multimediakommunikation**
4. Auflage, Springer Verlag, Peter Bocker
Berlin, Heidelberg, New York 1997

Index

Fujitsu Siemens Computers GmbH
User Documentation
33094 Paderborn
Germany

Fax: (++49) 700 / 372 00001

email: manuals@fujitsu-siemens.com
http://manuals.fujitsu-siemens.com

Comments
Suggestions
Corrections

Submitted by

Comments on CMX/CCP V6.0
ISDN Communication

Fujitsu Siemens Computers GmbH
User Documentation
33094 Paderborn
Germany

Fax: (++49) 700 / 372 00001

email: manuals@fujitsu-siemens.com
http://manuals.fujitsu-siemens.com

Comments
Suggestions
Corrections

Submitted by

Comments on CMX/CCP V6.0
ISDN Communication

Information on this document

On April 1, 2009, Fujitsu became the sole owner of Fujitsu Siemens Computers. This new subsidiary of Fujitsu has been renamed Fujitsu Technology Solutions.

This document from the document archive refers to a product version which was released a considerable time ago or which is no longer marketed.

Please note that all company references and copyrights in this document have been legally transferred to Fujitsu Technology Solutions.

Contact and support addresses will now be offered by Fujitsu Technology Solutions and have the format *...@ts.fujitsu.com*.

The Internet pages of Fujitsu Technology Solutions are available at *http://ts.fujitsu.com/*...
and the user documentation at *http://manuals.ts.fujitsu.com*.

Hinweise zum vorliegenden Dokument

Zum 1. April 2009 ist Fujitsu Siemens Computers in den alleinigen Besitz von Fujitsu übergegangen. Diese neue Tochtergesellschaft von Fujitsu trägt seitdem den Namen Fujitsu Technology Solutions.

Das vorliegende Dokument aus dem Dokumentenarchiv bezieht sich auf eine bereits vor längerer Zeit freigegebene oder nicht mehr im Vertrieb befindliche Produktversion.

Bitte beachten Sie, dass alle Firmenbezüge und Copyrights im vorliegenden Dokument rechtlich auf Fujitsu Technology Solutions übergegangen sind.

Kontakt- und Supportadressen werden nun von Fujitsu Technology Solutions angeboten und haben die Form *...@ts.fujitsu.com*.

Die Internetseiten von Fujitsu Technology Solutions finden Sie unter *http://de.ts.fujitsu.com/*..., und unter *http://manuals.ts.fujitsu.com* finden Sie die Benutzerdokumentation.