Prime

X.400 API Development Kit Administrator's Guide

Release 1.0

DOC11232-1LA

X.400 API Development Kit Administrator's Guide

First Edition

Peter Hassall and Liz Parsons

This book documents the use of
The Prime X.400 API Development Kit
at Release 1.0.

Prime Computer, Inc., Prime Park, Natick, MA 01760

PRINTING HISTORY

First Edition (DOC11232-1LA) October, 1988 for Release 1.0

CREDITS

Design: Authors
Editorial: John Wells
Project Support: Alan Mynard, Pete Lynch, Brian Etherington, Peter Kolsti
Illustration: Kevin Maguire
Document Preparation: Kevin Maguire
Production: Prime Technical Publications production unit

HOW TO ORDER TECHNICAL DOCUMENTS

To order copies of documents, or to obtain a catalog and price list:

United States Customers	*International*
Call Prime Telemarketing, toll free, at 1-800-343-2533, Monday through Friday, 8:30 a.m. to 5:00 p.m. (EST).	Contact your local Prime subsidiary or distributor.

CUSTOMER SUPPORT

Prime provides the following toll-free numbers for customers in the United States needing service:

1-800-322-2838 (Massachusetts)
1-800-541-8888 (Alaska and Hawaii)
1-800-343-2320 (within other states)

For other locations, contact your Prime representative.

SURVEYS AND CORRESPONDENCE

Please comment on this manual using the Reader Response Form provided in the back of this book. Address any additional comments on this or other Prime documents to:

Technical Publications Department
Prime Computer, Inc.
500 Old Connecticut Path
Framingham, MA 01701

CONTENTS

ABOUT THIS BOOK

The X.400 API Development Kit Administrator's Guide is a reference and guide to administering and controlling Prime X.400 on a system or network. It gives an overview of X.400 protocols, introduces Prime X.400, and describes in detail how to configure the Prime X.400 logical model.

The book also describes the ADMIN_X400 operator utility that you use to start and stop Prime X.400 on the system, and the ADMIN_X400 subcommands that you use to monitor and control Prime X.400 operation.

Chapter Contents

Chapter 1 — Introduction to Prime X.400, provides a general introduction to X.400, Prime X.400 and Prime X.400 configuration. It also describes Prime X.400 security, and introduces the operator commands.

Chapter 2 — Prime X.400 Configuration, is a guide and reference to the principles and practice of configuring Prime X.400 on a network. It describes the CONFIG_X400 configurator command, and contains details of the configurator menus and forms.

Chapter 3 — Operation and Monitoring, describes the operator command ADMIN_X400, starting and stopping Prime X.400, and the display and control subcommands.

Chapter 4 — CONFIG_X400 EXAMPLES, provides a tutorial to configuring Prime X.400 on your system or network.

Appendix A — Installation and Diagnostics, explains how to install Prime X.400, describes the PRIME_X400* product directory, and outlines how to use the error logs and journals to monitor X.400 sessions and diagnose problems.

Appendix B — Error Messages, lists and describes the hexadecimal error codes produced by Prime X.400.

Related Documentation

Companions to this book are:

- *X.400 API Development Kit Programmer's Guide (DOC11234-1LA)*

Other Prime manuals which you may find useful for reference are:

- *Network Planning and Administration Guide (DOC7532-3LA)*
- *Operator's Guide to Prime Networks (DOC10114-LA)*
- *System Administrator's Guide Vol. 1, System Configuration (DOC10131-1LA)*

Other manuals which you may find useful for reference are:

- *CCITT Red Book Volume VIII Fascicle VIII.7, Recommendations X.400 - X.430*

Prime Documentation Conventions

The following conventions are used in command formats, statement formats, and in examples throughout this document. Examples illustrate how you use these commands and statements in typical applications.

Convention	*Explanation*	*Example*
UPPERCASE	In command formats, words in uppercase indicate the names of commands, options, statements, and keywords. Enter them in either uppercase or lowercase.	DISPLAY-USER
lowercase	In command formats, words in lowercase indicate variables for which you must substitute a suitable value.	CONFIG_X400 filename
Abbreviations in option descriptions	If an uppercase word in a command format has an abbreviation, the name and abbreviation are placed within braces.	{ **-HELP** / **-H** }
Underscore in examples	In examples, user input is underscored but system prompts and output are not.	OK,display-user user=all
Angle brackets in messages < >	In messages, text enclosed within angle brackets indicates a variable for which the program substitutes the appropriate value.	<filename> not found.
Boldface	When they first appear in text, new terms are entered in boldface.	**applications**
Italics	In text, italics indicate variable user input or emphasis. Where Prime documentation is referred to in text, the title of the manual is entered in italics.	*pathname* the *default* file *Prime X.400 API Guide*
Monospace	User examples and program listings are displayed in monospace.	

PART I
OVERVIEW AND REFERENCE

1

INTRODUCTION TO PRIME X.400

This chapter gives an overview of the X.400 message handling system, introduces the Prime X.400 product, and outlines the process of Prime X.400 configuration. It also introduces user security on Prime X.400 services, and the ADMIN_X400 operator command.

Introduction

Prime X.400 is a set of communication software services that allow Prime systems to connect to, and interchange data with, X.400 applications on other networks. It contains the support services for all X.400 applications, and forms the basis on which electronic mail applications can be designed and built for Prime systems.

Prime X.400 implements the CCITT X.400-series recommendations for message handling systems. For details of these recommendations, refer to the *CCITT Red Book, Volume VIII Fascicle VIII.7* covering CCITT Recommendations X.400-X.430.

Overview of X.400

X.400 is a series of protocols that define a store-and-forward Message Handling System (MHS) for the exchange of messages between computer network users. It addresses primarily the requirements of electronic mail applications. X.400 is a Message Handling System application implemented in layer 7 of the OSI Seven Layer Reference Model. Figure 1-1 illustrates the OSI Reference Model.

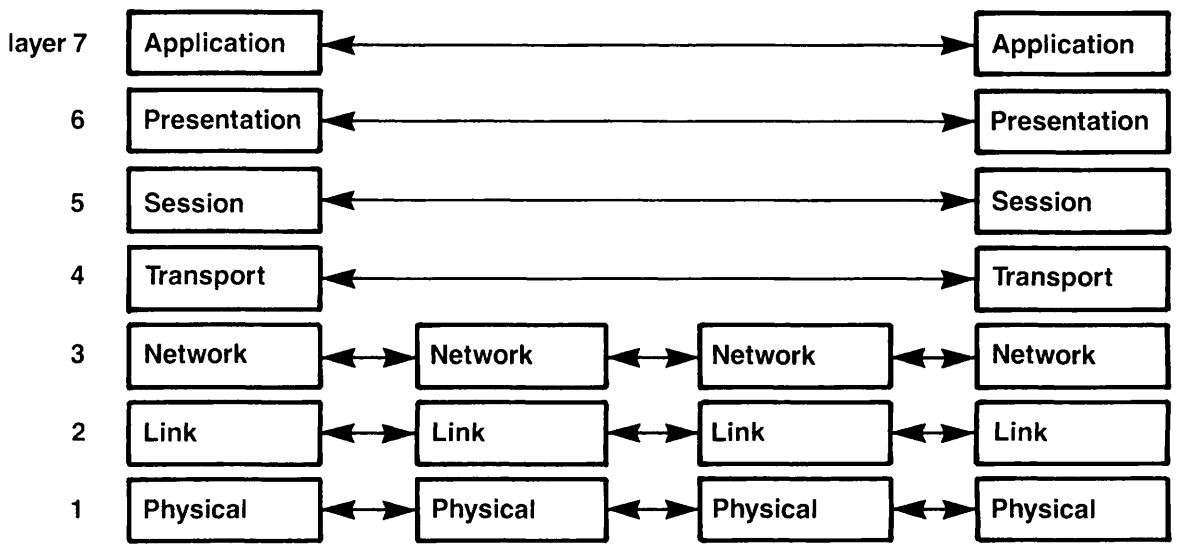

FIGURE 1-1. The OSI Reference Model

The X.400 Model

The X.400 series of definitions and protocols define a network logical model to which all X.400-compatible message handling systems must conform. The model consists of two types of software processes, known as User Agents (UAs) and Message Transfer Agents (MTAs).

User Agents provide the link between users and MTAs. UAs are processes that interact with the sender, construct messages for submittal to MTAs, and display them to recipients at the target node. User Agents are implemented by mail applications.

Message Transfer Agents are the store-and-forward nodes on an X.400 network. They act as relay points for the exchange of messages across the network, cooperating with each other to ensure delivery. MTAs act as the intermediaries between User Agents, determining destinations, controlling routing, delivering messages, and signalling errors.

X.400 User Addresses

Users are identified on X.400 networks by **Originator/Recipient** (O/R) Addresses. These are unique addresses assigned by network administrators, in cooperation with X.400 regulatory bodies. X.400 permits two formats for O/R addresses, as follows:

- A multi-component name
- An X.121 Address

The multi-component name is the most common form, and is used by Prime X.400. Components include the country where the network operates, the names of administration domains, organizations and organizational units, and personal names. Some components are allocated by the X.400 international regulatory bodies, some by the public service carriers in a particular country, some by suppliers of X.400 software, and some by administrators of individual networks.

For details of X.400 address components, and how Prime X.400 uses them, see Chapter 2, PRIME X.400 CONFIGURATION.

Prime X.400 Concepts

This section describes how Prime X.400 implements the X.400 design principles and protocols.

The Prime X.400 Logical Network

In accordance with the X.400 model, Prime X.400 consists of Message Transfer Agents that act as store-and-forward nodes for the exchange of messages between X.400 over the network, and User Agents that interface with users to provide the message transfer service. In Prime X.400, User Agents are implemented by Prime X.400 applications that use the Prime X.400 services provided by the Application Programming Interface (API).

Figure 1-2 illustrates the main components of the logical Prime X.400 network.

Local and Remote MTAs

Message Transfer Agents can be local or remote.

Local MTAs are those that are controlled as part of one administrative unit. Typically they would be configured on a group of systems controlled by a single administration, where there is no conflict of user names. Within such a unit, the administrator needs access to the Prime X.400* directories and configuration files on all systems, in order to define or modify the configuration. The administrator of the unit would be expected to have this level of access.

Data that administrators supply for local MTAs are their locations on the physical network, passwords, the logical links **(associations)** between them and with remote MTAs, X.400 protocol parameters, and O/R addresses for local users.

The Prime X.400 configuration utility automatically inserts default values for much of the link and protocol data, so that you need only specify them if the network uses uncommon or special protocols.

Remote MTAs are defined as being on systems, or groups of systems, that are under the control of other administrators. Data that administrators must supply about remote MTAs with which they wish to communicate, are their network addresses, some X.400 protocol parameters, and the O/R address space of the remote users.

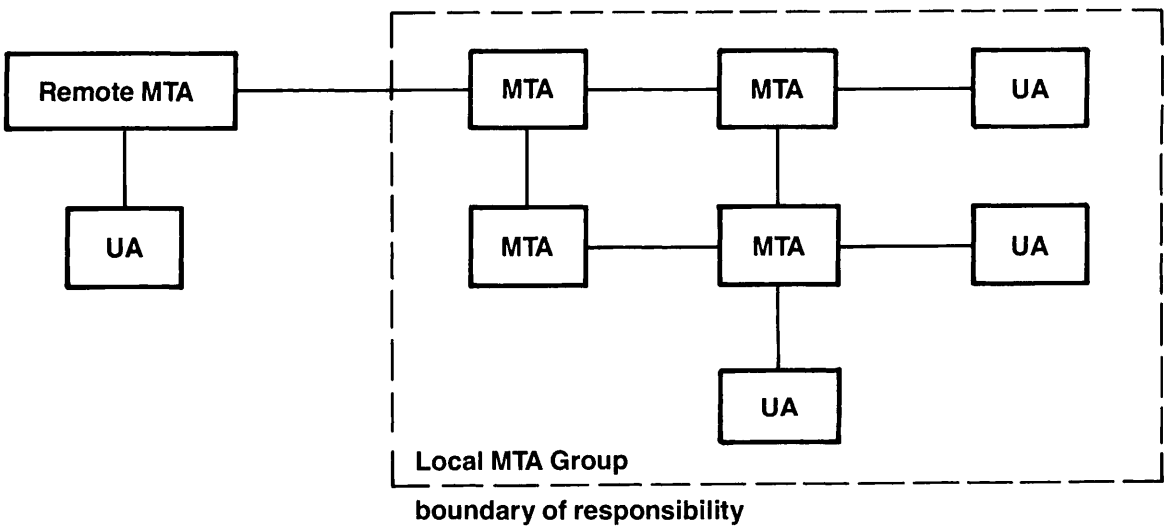

FIGURE 1-2. The Prime X.400 Logical Network

Prime X.400 Configuration

Prime X.400 Configuration is the procedure the administrator goes through to specify the parameters that Prime X.400 needs to operate on the local network, and to communicate reliably with X.400 applications on other networks. It defines local and remote MTAs on the network, assigns user addresses to users, both local and remote, and allows protocols to be specified, if necessary, for communicating with specific MTAs.

The CONFIG_X400 Command

The CONFIG_X400 command allows the administrator to create or modify a configuration, perform verification checks, and save the configuration, within a single terminal session. CONFIG_X400 creates a configuration file from which Prime X.400 builds its routing tables when it is started on the system.

The CONFIG_X400 command invokes an interactive environment that uses a hierarchy of selection menus and data input screens to help you define your configuration. For further details of the interactive subsystem, see Chapter 2, Prime X.400 Configuration, and Chapter 4, CONFIG_X400 EXAMPLES.

Control and Monitoring

The ADMIN_X400 command provides for server startup, shutdown, and for monitoring the activity of Prime X.400 on the system. Startup and shutdown are performed by command-line options, and monitoring facilities are provided within a subcommand environment. Subcommands allow you to display the status of users, MTAs, and message queues, to stop and reconfigure Prime X.400 on the system, and to enable error display.

For details of the subcommands, see Chapter 3, OPERATION AND MONITORING.

2

PRIME X.400 CONFIGURATION

This chapter is a guide and reference to configuring Prime X.400 on a network. It introduces Prime X.400 configuration and X.400 user addressing, and describes the CONFIG_X400 interactive configurator in detail, with illustrations of screen forms.

Introduction

Prime X.400 Configuration is the process by which you assign MTAs to physical locations on the network, set up the logical links between local MTAs, and between local and remote MTAs, and define local and remote user addresses.

Prime X.400 configuration produces a file that Prime X.400 uses to construct its routing and directory tables that are built when Prime X.400 is started on the system.

Configurations are defined using the CONFIG_X400 command. CONFIG_X400 is a screen-based interactive subsystem that allows you to define your configuration step by step, verify it for correctness, and save it to disk.

Configuration Data

The main categories of information to supply when configuring Prime X.400 are as follows:

- The number of local MTAs on your network, and the protocols under which they operate.

  You can assign MTAs to each node on the network, to a single node, or to selected nodes. However, a single node can contain only a single MTA.

- Local user addresses, and the MTAs to which they are attached.

  Each user is attached to a specific MTA, which acts as the collecting and distribution point for all X.400 messages to and from that user.

  A user's MTA can be the node where the user logs in, or any convenient node accessible via the local network.

- The names of remote MTAs with which you wish to communicate, protocols for communicating with them, and the user address domain that they control.

Prime X.400 Configuration Concepts

This section introduces some configuration concepts that you, the administrator, should be familiar with before configuring Prime X.400 on your system or network.

Local MTAs

The Prime X.400 administrator is responsible for configuring Prime X.400 on a single node or a group of local nodes, such as a local network, which he or she administers. The MTA group that the administrator defines is known as the local MTA group.

MTAs in the same local group share a set of default attributes, such as the type of logical link **(association)** between them, the default address space **(domain)**, and inter-MTA passwords.

You can only define MTAs on nodes where you have access to the PRIME_X400* directory to create and modify the configuration file. In specific terms, you must have DALURW (Delete, Add, List, Use, Read and Write) access to PRIME_X400* on the system.

Information you must supply for local MTAs are: their network addresses, the link type (association) between them, passwords, service protocols and routing information, and users' X.400 addresses and MTA attachments.

Remote MTAs

Remote MTAs are MTAs on nodes outside your immediate control. They act as gateways to users in other X.400 domains.

Information you must supply for a remote MTA is: its network address, protocols for communicating with the local MTA group, and the remote user address space (domain) associated with it. This information must be exchanged with the remote administrator.

Associations

Associations are the logical connections between MTAs that comprise the X.400 network; they are the communication channels through which data transfer takes place.

Associations can be either permanent or temporary. Permanent associations are maintained at all times, even across failures of the underlying network. Temporary associations are created in response to specific user requests (such as the sending of a message), and are subject to a timeout period for inactivity, but are not maintained across network failures. By default, all MTAs in a local configuration are adjacent, and are mutually interconnected by a single temporary association.

Subnetworks

Prime X.400 allows you to define sets of network data and protocols to identify networks of any type that may be encountered when connecting to other X.400 products and applications. These are known as **subnetworks.**

Subnetworks are defined separately for a particular configuration. Once the subnetwork is defined, you can identify the the network protocols for a specific MTA simply by giving a subnetwork name.

Data that you specify when defining a subnetwork are parameters for lower-level protocols in the OSI model. For brief descriptions of the parameters see Figure 2-8, and for further details consult the relevant ISO documentation.

Domains

Prime X.400 domains are X.400 name spaces through which users can be addressed. Each domain is described by a set of high-level O/R address components (see below). Local domains are those associated with local MTAs, and remote domains, are those associated with remote MTAs.

User Addresses

Users are identified within Prime X.400 directly by unique X.400 O/R addresses (see Chapter 1, Introduction to Prime X.400).

The address can take three forms; in all three forms *Country* and *ADMD* are mandatory.

The three forms are:

1. Country and ADMD plus one or more of:

   PRMD
   Personal Name
   Organization
   Organizational Unit

2. Country and ADMD plus a UA Unique Numeric Identifier
3. Country and ADMD plus an X.121 Address

Form 1 is most commonly used and is implemented by Prime X.400 when sending mail. All three forms are supported for received and relayed mail.

Address Components

O/R address components supported by Prime X.400 are listed below.

Country
Administration Management Domain (ADMD)
Private Management Domain (PRMD)
Organization
Organizational Unit(s)
Personal Name:

Surname
Given Name
Initials
Generation Qualifier

Domain Defined Attributes
X.121 Address
Terminal ID
UA Unique ID

For details of address components and their meaning, see later in this chapter.

Name Allocation

Some components of the domain may be set by the regulatory authorities in your country, and the public carriers to which you connect. The components to which this is likely to apply are Country, ADMD, and PRMD.

Country is allocated by the X.400 international controlling authorities. It represents the country in which the network operates. It may be either a three digit code as defined by CCITT X.121, or a two letter code defined by ISO 3166/ALPHA-2.

ADMD is an X.400 administrative domain within the country, for example the X.400 service provided by a public carrier. For private mail applications that are not linked to other carriers, ADMD is not used.

PRMD is a private X.400 administration domain, often a supplier's name. If you are connecting to X.400 through a public carrier, you will probably have to negotiate the name of the PRMD with that supplier. If you are connecting to a private mail application, you should be able to choose your own PRMD name.

For example, the following set of names could be allocated to Prime networks within the UK:

Country : *GB*

ADMD : *Gold 400*

PRMD : *Prime*

Other components can be defined by administrators to suit their administrative structures.

For example, an administrator of a PRIMENET™ network on several sites could assign Organization Names to individual sites, leaving the administrators of those sites to define their own Organizational Unit names for departments within the site. Hierarchical naming schemes of this kind are likely to be the most commonly encountered in practice, but any naming system can be used, including nonhierarchical systems.

Configuration Planning

Although Prime enforces no topology on you, Prime X.400 configuration does require some planning and forethought. This section introduces some of the issues you should address before configuring MTAs and users on your system or network.

How Many Configurations?

Prime X.400 is configured within existing administrative units, where administrators have access to system directories, and control user access to the system. Single Prime X.400 configurations should be implemented where user names can be guaranteed to be unique. This is the recommended practice, as it avoids conflicts between mail users with the same login ID.

Multiple configurations are only necessary when the user population is large, or where user names cannot be guaranteed unique. For many installations, a single configuration will be sufficient.

How Many MTAs?

MTAs are the store-and-forward nodes of an X.400 network. Each MTA stores the information it needs to forward messages to the next node.

You can configure one MTA per PRIMENET node, a single MTA for the whole of your PRIMENET network, or any number between. If an MTA serves more than one node, you must ensure that the PRIME_X400* directory on the node where the MTA is configured (which contains the routing table), is visible to all the other nodes served by the MTA.

If you are unsure, configure one MTA per node.

O/R Addresses

MTAs define specific addressing spaces on the X.400 network. You can configure them to suit your installation, and the mail needs of your users.

The domain (Country, ADMD, and PRMD) will be the same for all local MTAs. Below the domain level, Organization and Organizational Unit names may be imposed by company structure or national agencies. There are cases where the Organization name must be the nationally-registered name of the company and the administrator should be aware of this.

Routing

Routing in Prime X.400 is controlled by a fixed strategy. Components of the O/R address are tested in a fixed order. Local addresses are matched first.

Routing to remote users is controlled by the National Bureau of Standards (NBS) routing class of the MTA. Class 1 MTAs can route on Country, ADMD, PRMD, and Organization, class 2 on Organizational Unit, and class 3 on Personal Names. Class 3 MTAs are less likely to cause ambiguous routing, but consume more storage for their routing tables.

The CONFIG_X400 Command

You configure Prime X.400 using the CONFIG_X400 command. This command allows you to create and maintain information about the Prime X.400 configuration on your network.

The configuration is maintained in an ASCII file. An in-memory version of the configuration is built when Prime X.400 is started on the system.

The syntax of the CONFIG_X400 command is as described in the following section.

```
                filename -TERMINAL_TYPE terminaltype
▶ CONFIG_X400   -HELP [ -NO_WAIT ]
                -USAGE
```

Options are described below.

filename

The pathname of an existing configuration file or the name of a new file to be created. If you do not specify a suffix, the suffix .CONFIG is added automatically.

To *modify* an existing configuration, specify an existing configuration file.

To *create* a new configuration, specify a filename. If the file does not already exist, a new file is created to contain the configuration.

Filename is optional. If you do not specify a filename, your configuration is written to the file PRIME_X400*>X400.CONFIG by default.

{ -TERMINAL_TYPE | -TTP } terminaltype

Specifies the terminal type you are using. Example terminal types are:

PST100
PT200
PT200-C

(Refer to the Priforma Manual).

If you do not specify a terminal type, the command uses the type you have defined in the global variable .TERMINAL_TYPE$. If you do not specify a terminal type, and you have not set the terminal type global variable, the command aborts with an error message.

There is no default terminal type.

{ -HELP } [{ -NO_WAIT }]
{ -H }   [{ -NW }   ]

Explains how to use the command. This option cancels any other options on the command line. If you specify -NO_WAIT, the display is not paginated at your terminal. The same information is available through the PRIMOS® HELP subsystem.

-USAGE Gives you the command syntax in brief.

When you invoke the CONFIG_X400 command, information about the configuration input file is displayed, as in Figure 2-1.

```
Config File: PRIME_X400*>PRIME_X400.CONFIG

Revision Number:
Last Updated   :
Updated by user:
Updated on node:

Comment:
```

FIGURE 2-1. Configuration File Information

The CONFIG_X400 Subsystem

CONFIG_X400 is a screen-based interactive subsystem through which you define your configuration step by step, verify its correctness, and save it to disk. You select options, and input data and parameters, through a linked hierarchy of screen forms that you navigate using special function keys.

The full hierarchy of CONFIG_X400 functions and subfunctions is illustrated in Figure 2-2.

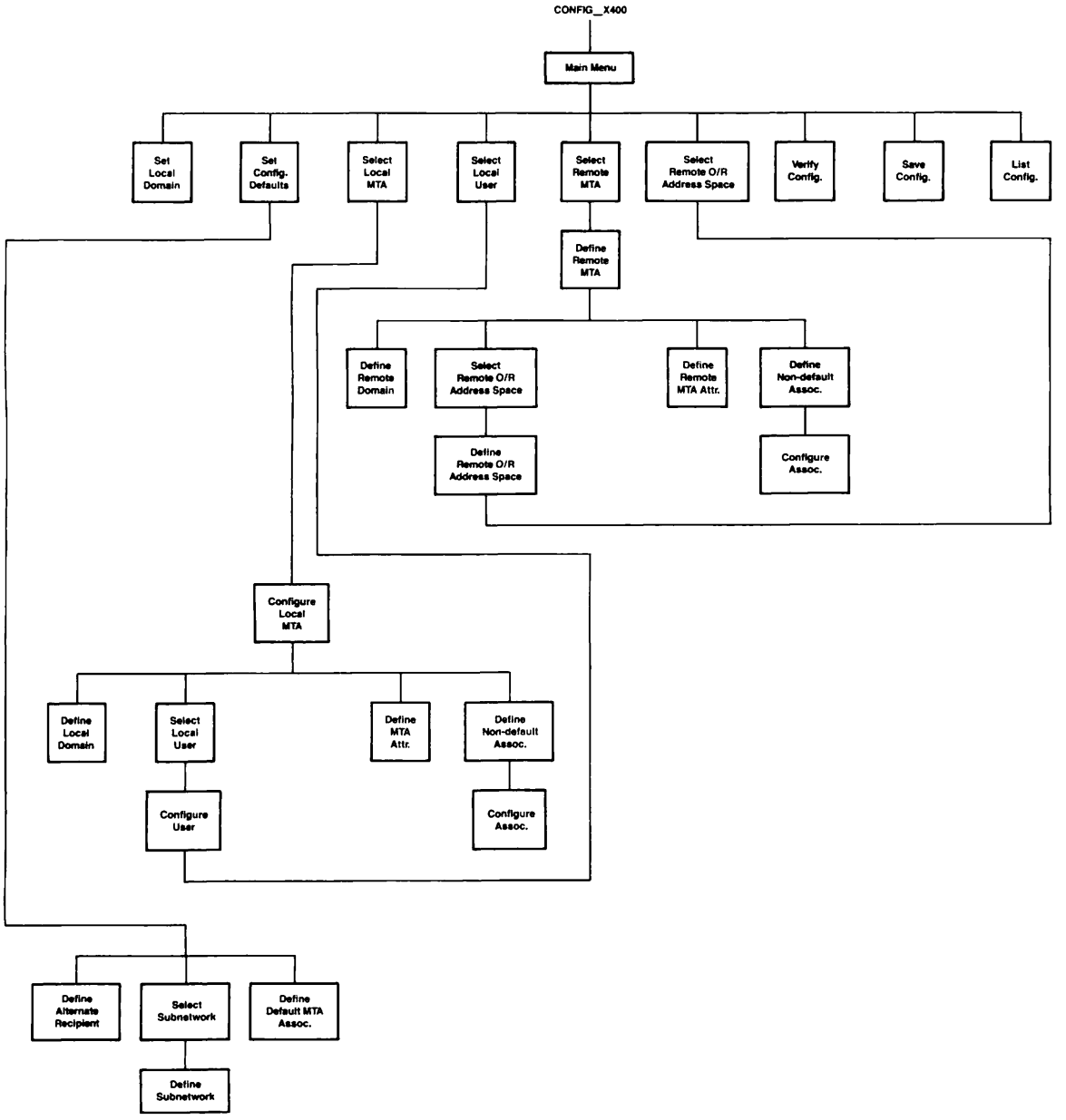

FIGURE 2-2. CONFIG_X400 Menu Hierarchy

Function Keys

Function keys allow you to perform operations such as moving between fields, selecting options, and committing changes. The functions that are available, and the keys to use to perform these functions on the PT200™ and PST100™ keyboards are listed in table 2-1.

Functions that are available on specific screen forms, and the keys that correspond to them, are also displayed at the bottom of screen.

TABLE 2-1. CONFIG_X400 Function Keys

Function	*PT200*	*PST100*	*Operation*
TAB	[TAB]	[TAB]	Move to next field
BACK TAB	[Back Tab]	[BACK TAB]	Move to previous field
TOPSCR	[F8]	[F8]	Return to main menu
TRANSMIT	[Enter]	[PF10]	Commit a change
HELP	[Help]	[HELP]	Display Help
PRINT-SCREEN	[Prt Scn]	Shift [AUX SEND]	Print screen, or write to file
EXIT	[Cancel]	[PF14]	Return to previous screen, quit program, or clear error message
ADD	[F3]	[F3]	Add a new entry to a list
REMOVE	[F4]	[F4]	Remove an entry from a list
PRVPAGE	[F5]	[F5]	Display the previous page of a multi-page form
NXTPAGE	[F6]	[F6]	Display the next page of a multi-page form

Screen Forms

Screen forms are areas of the screen surrounded by reverse video. There are two types: *menus*, and *data input forms*.

Menus: Selection menus consist of a list of options from which you select the one you require; data input forms consist of labeled input fields where you enter data, arguments, and parameters.

To select an option from a menu, position the cursor on the option, and press the TRANSMIT key. Alternatively, enter the number of the option you require, and press the TRANSMIT key.

Data Input Forms: Data input forms consist of collections of fields where you specify data such as name identifiers and protocol parameters. To input data, type within the defined field.

Fields are defined on the form by underlines or by reverse video. The length of the underline indicates the maximum size of the field.

To move between screens, forms, and fields, use the function keys defined in Table 2-1.

Selecting Entries From Help Screens

Entries for some fields can be selected from Help screens. To display Help for a particular field, position the cursor on the field, and press the HELP key. Selectable entries, if available, are listed on the Help screen.

To select an entry from the Help display, position the cursor on the entry, and press TRANSMIT.

Default Configuration Data

CONFIG_X400 provides defaults for many Prime X.400 configuration parameters. Where defaults are present, the field is prefilled with the default value.

The default parameters operate for most installations. Only change them if you have special reasons for configuring Prime X.400 in a different way.

CONFIG_X400 Main Menu

The CONFIG_X400 Main Menu lists the main configurator functions, and allows you to select the one you require.

The Main Menu functions allow you to specify categories of information to configure, verify, save, and list the configuration.

Main categories of information to configure are:

- Default local domain
- Configuration defaults

- Local user addresses
- Non-default local domains
- Remote MTA domains
- Remote MTA Attributes
- Remote user addresses

The CONFIG_X400 Main Menu is illustrated below:

```
) CONFIG X400 - Main Menu (

Select configuration option:

        1.  Set Local Domain
        2.  Set Configuration Defaults

        3.  Configure Local MTAs
        4.  Configure Local Users

        5.  Define Remote MTAs
        6.  Define Remote O/R addresses

        7.  Verify Configuration
        8.  Save Configuration
        9.  List Configuration

Press:  <TRANSMIT>  to invoke selected option.
        <EXIT>      to exit the configurator.
```

FIGURE 2-3. CONFIG_X400 Main Menu

CONFIG_X400 Option 1 - Set Local Domain

The `Set Local Domain` option in this Main Menu (Figure 2-3), allows you to define a default local domain. A local domain is a default address space for X.400 users within your configuration, containing Country, ADMD, and PRMD names. The addresses of all local users contain this set of names unless you specify otherwise.

The domain name is also used to identify the MTA, to any remote MTAs with which it is associated.

The Set Local Domain form is illustrated in Figure 2-4.

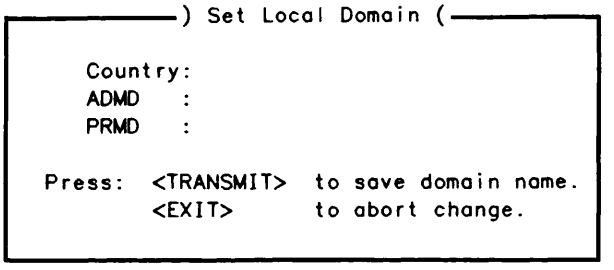

FIGURE 2-4. Set Local Domain Form

Define the local configuration domain by specifying Country, ADMD and PRMD.

Field	*Description*
Country	The country in which the network operates. This is defined by international regulatory authorities. Specify either: • A three-digit code as defined by CCITT X.121 (for example, the UK code is 234, the US code is 311). • A two-letter code as defined by ISO 3166/ALPHA-2 (for example, GB, US).
ADMD	The main administration domain with which you are associated. If you are connecting to a public service mail system, use the name they give you. If you are connecting to a private mail application, choose one by bilateral agreement, or leave the field blank. The ADMD can contain a maximum of 16 characters.
PRMD	The private administration domain with which you wish to be associated. If you are associated with an ADMD, the PRMD may be defined by the administration of the ADMD. If you are connecting to an independent mail application, agree the name with the administrator responsible for the application. The PRMD can contain a maximum of 16 characters.

CONFIG_X400 Option 2 - Set Configuration Defaults

The Set Configuration Defaults option in the Main Menu (Figure 2-3), allows you to set configuration defaults. You can specify a default recipient of undelivered mail, defaults for subnetwork definitions, and associations between MTAs.

When you select the Set Configuration Defaults option, the Set Configuration Defaults menu is displayed, as in Figure 2-5.

```
) Set Configuration Defaults (

Select option:

          1.  Alternate Recipient
          2.  Network Parameters
          3.  Associations

Press:  <TRANSMIT>  to invoke selected option.
        <EXIT>      to return to main menu.
```

FIGURE 2-5. Set Configuration Defaults Menu

Alternate Recipient

When you select option 1, Alternate Recipient, on the Set Configuration Defaults menu, the Define Alternate Recipient form is displayed, as in Figure 2-6.

```
) Define Alternate Recipient (

Recipient of all undeliverable mail:

   Mail Logon ID:
   MTA Name     :

Press:  <TRANSMIT>  to save details.
        <EXIT>      to abort change.
```

FIGURE 2-6. Define Alternate Recipient Form

Fields on the Set Alternate Recipient form are described below.

Field	*Description*
Mail Logon ID	The mail ID of a local user who will receive all non-deliverable mail.
MTA Name	The MTA to which the alternate recipient is attached may be local or remote. For remote MTAs, the Mail Logon ID should be left blank.

Network Parameters

Prime X.400 subnetworks are sets of protocol data and parameters that define specific subnetworks and subnetwork types.

Note

Unless you specify a subnetwork, the Network Provider is assumed to be X25, and the X25 Year 1980; other subnetwork parameters are left undefined.

When you select option 2, `Network Parameters`, on the Set Configuration Defaults menu (Figure 2-5), the Select Subnetwork form is displayed, as in Figure 2-7.

```
) Select Subnetwork (

Select Subnetwork Definition (SND):




Press:  <TRANSMIT>  to select SND at cursor.
        <EXIT>      to abort function.
        <ADD>       to add a new SND.
        <REMOVE>    to delete SND at cursor.
```

FIGURE 2-7. Select Subnetwork Form

You can select a definition to modify, add a new definition, or remove a definition.

When you select a definition to modify, or add a new one, the Define Subnetwork form is displayed, as in Figure 2-8.

```
) Define Subnetwork (

Subnetwork Name:

Network Provider:X25      X25 Year:       Fast Select:

Destination Address (X121):
Destination Protocol ID   :
Destination Facility      :

Source Address (X121):
Source Protocol ID   :

Press:  <TRANSMIT>  to save definition.
        <EXIT>      to abort change.
```

FIGURE 2-8. Define Subnetwork Form

Subnetwork definition fields are described in the following list.

Field	*Description*
Subnetwork Name	Your name for the subnetwork.
X25 Year	Enter the X25 year; for example, 1980.
Fast Select	Selects the X25 Fast Select facility. Enter YES or NO. This should normally be entered as NO, which is the default.
Destination Address (X121)	The X121 address on outgoing calls. By default, this is computed from the remote Network Service Address (NSAP).
Destination Protocol ID	The X25 protocol ID used by the remote MTA. Four bytes, entered as ASCII hexadecimal digits.
Destination Facility	The X.25 Facilities to propose when making an X.25 call to the remote system which operates the Remote MTA. For details of how to encode X.25 Facilities, refer to the *Primenet Programmer's Guide.* Facilities are not normally required for correct operation.
Source Address (X121)	The X121 address for incoming calls. This is normally the same as one of the local Primenet addresses. Source Address should be distinct from any subaddress used for PRIMENET Route-through (see PRIMENET Guide).

Source Protocol ID — The X25 protocol ID used by the local MTA. Enter as four bytes in ASCII hexadecimal format.

Associations

When you select option 3, `Associations`, on the Set Configuration Defaults menu (Figure 2-5), the Define Default MTA Associations form is displayed, as in Figure 2-9.

```
) Define Default MTA Associations (

Temporary:                  Local              Remote
   Number of Associations:
   Subnetwork Definition :
   Timeout (minutes)     :

Permanent:
   Number of Associations:
   Subnetwork Definition :

Press:  <TRANSMIT>  to save default values.
        <EXIT>      to abort change.
```

FIGURE 2-9. Define Default MTA Associations Form

Specify the number of temporary and permanent associations for local and remote MTAs, and the name of the previously defined subnetwork. For temporary associations only, specify the inactivity timeout period (temporary associations are only maintained while messages are being exchanged).

The maximum number of associations of each of the four types is 999, and the maximum timeout period is 999 minutes.

CONFIG_X400 Option 3 - Configure Local MTAs

The `Configure Local MTAs` option in the Main Menu (Figure 2-3), allows you to specify network addresses and protocols for MTAs in your local group, and configure users on them.

Data that you specify when you define local MTAs is as follows:

- MTA names, network addresses, routing and protocol information
- MTA attachments and O/R Addresses of local users

When you select the `Configure Local MTAs` option, the Select Local MTA form is displayed, as in Figure 2-10.

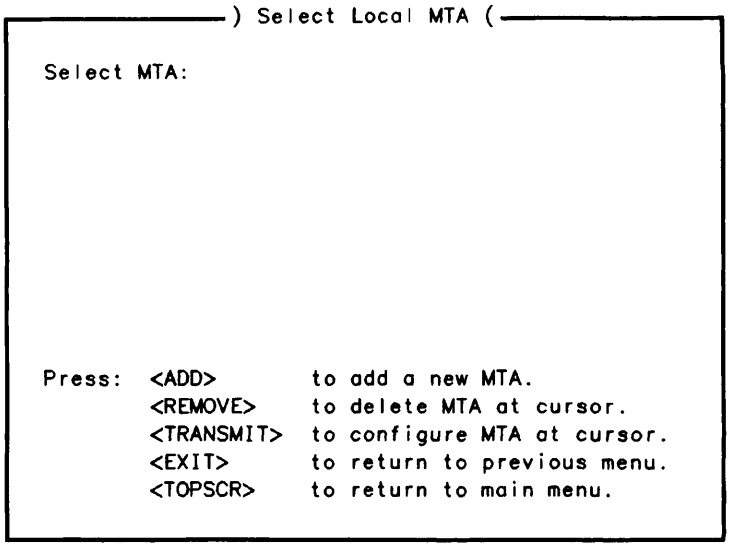

FIGURE 2-10. Select Local MTA Form

When you have selected an MTA to modify, or added a new one, the Configure Local MTA menu is displayed, as in Figure 2-11.

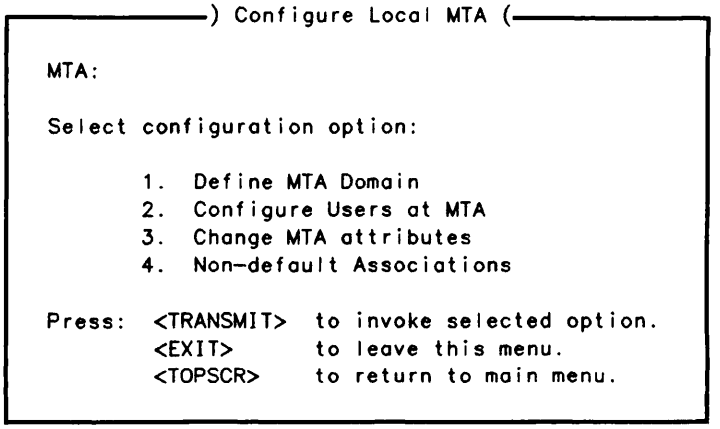

FIGURE 2-11. Configure Local MTA Menu

Define MTA Domain

If you select option 1, `Define MTA Domain`, on the Configure Local MTA menu (Figure 2-11), the Define Local Domain form is displayed, as in Figure 2-12.

```
) Define Local Domain (

MTA Name:

Country:
ADMD    :
PRMD    :

Organization:

Organizational Units:




Press:  <TRANSMIT>  to save details.
        <EXIT>      to abort change.
```

FIGURE 2-12. Define Local Domain Form

Configure Users at MTA

If you select option 2, `Configure Users at MTA`, on the Configure Local MTA menu, the Select Local User form is displayed, as in Figure 2-13.

Bulk Loading Users From an Existing List

Users on existing mail lists can be included in the configuration using the bulk loading procedure. Briefly, this involves editing the existing list to produce user records of the correct format, appending the edited list to the configuration file, and invoking the configurator to save the new configuration.

For an example of how to use the bulk loading procedure, see Chapter 4, CONFIG_X400 EXAMPLES.

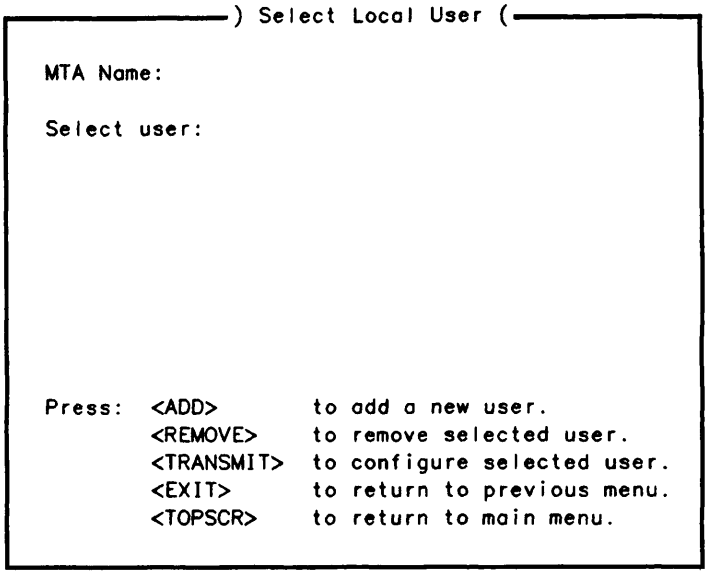

FIGURE 2-13. Select Local User Form

When you have selected a user to modify, the Configure User form pages 1, 2, 3, and 4 are displayed in sequence.

Pages 1 and 2 allow you to specify the O/R address components, as in Figure 2-14.

Page 3 allows you to control access to Prime X.400 resources, and page 4 allows you to select data types supported by the user. Pages 3 and 4 are illustrated in Figure 2-15.

```
                    ) Configure User (       ) Page 1 of 4 (

MTA:

Mail Logon ID:

O/R Address:
   Personal Name:
     Surname :                        Given Name:
     Initials:              Generation Qualifier:

   Organization Name:
     Organization:
     Units:

   Country:       ADMD:                    PRMD:

Press:  <TRANSMIT>  to save user details.
        <EXIT>      to abort changes.
        <NXTPAG>    to view next page.
        <PRVPAG>    to view previous page.
```

```
                    ) Configure User (       ) Page 2 of 4 (

O/R Address (cont.):

  Domain Defined Attributes:

     Type       Value



Press:  <TRANSMIT>  to save user details.
        <EXIT>      to abort changes.
        <NXTPAGE>   to view next page.
        <PRVPAGE>   to view previous page.
```

FIGURE 2-14. Configure Local User Form, Pages 1 & 2

O/R address components and their meanings are described in the following list.

Country, ADMD, PRMD and Organization, each default to the values defined in the MTAs local domain name, if any.

Component	*Description*
Country	The country in which the network operates. This is defined by international control bodies. Specify either: • A 3-digit code as defined by CCITT X.121 (for example, the code for the UK is 234, for the US is 311) • A 2-letter code defined by ISO 3166/ALPHA-2 (for example, GB, US).
ADMD	Main administration domains within a country. If you are connecting to an X.400 service provided by a public carrier, use the name they give you. If you are connecting to a private mail application, either choose a name by bilateral agreement, or leave the field blank. The ADMD name can contain a maximum of 16 printable characters.
PRMD	Private administration domains within a country. For mail applications that connect to ADMDs, the PRMD is a matter of negotiation with the ADMD administrator. The PRMD name can contain a maximum of 16 printable characters.
Organization Name	A name assigned to an organization within a PRMD. It can contain a maximum of 64 printable characters.
Organizational Unit(s)	Names of units within an Organization. Up to 4 Organizational Units can be defined. Organizational Units can contain a maximum of 32 printable characters.
Personal Name	A collection of names that identify an individual mail service user: Surname : 40 characters Given Name : 16 characters Initials : 5 characters Generation Qualifier : 3 characters
Domain Defined Attributes	A group of attributes that are exclusive to X.400 management domains, either ADMDs, or PRMDs. Domain Defined Attributes can be used to map an existing mail service directory to the O/R addressing scheme. Attribute names can contain a maximum of 8 characters, and attribute values a maximum of 128 characters. Non-printing characters are allowed.

```
) Configure User (—) Page 3 of 4 (

User Access Control:

   Mail Logon ID:

   System Login ID/Group                Send  Receive



Press:  <TRANSMIT>  to save user details.
        <EXIT>      to abort changes.
        <NXTPAGE>   to view next page.
        <PRVPAGE>   to view previous page.
```

```
) Configure User (—) Page 4 of 4 (

User Supported Data Types:

        Undefined                 :
        Telex                     :
        Teleprinter (iA5)         :
        Group 3 Faxsimile         :
        Text Interchange Format 0:
        Teletex                   :
        Videotex                  :
        Voice                     :
        Simple Formatted Document:
        Text Interchange Format 1:

Press:  <TRANSMIT>  to save user details.
        <EXIT>      to abort changes.
        <NXTPAG>    to view next page.
        <PRVPAG>    to view previous page.
```

FIGURE 2-15. Configure Local User Form, Pages 3 & 4

User Access Control

By default, X.400 Mail Logon IDs are assumed to be the same as Primos User Logon IDs, so that users have access to both send and receive mail services.

If the X.400 Mail Logon ID is different to the Primos User Logon ID, or the access required is other than the default, then the Primos User Logon ID and mail service access rights for that mail user have to be explicitly stated.

User Supported Data Types

Specify the data types that the user can accept. The range of data types that are supported depends on the mail application or product that uses Prime X.400. (Specify YES or NO in each field).

Change MTA Attributes

If you select option 3, Change MTA attributes, on the Configure Local MTA menu, the Define MTA Attributes form is displayed, as in Figure 2-16.

```
) Define MTA Attributes (

MTA Name:

Password:                        NBS Routing Class:


Network Address            :
Transport Protocol Selector:

Reliable transfer Service (RTS):
      Checkpoint Size:                Window Size:

Session Protocol Data Unit (SPDU):
      Maximum transmit size:          Maximum receive size:

Press:  <TRANSMIT>  to save changes.
        <EXIT>      to abort the change.
```

FIGURE 2-16. Define MTA Attributes Form

Local MTA Attributes are described in the following list.

Attribute	*Description*
Password	The password that controls communication with the MTA. The MTA only accepts associations from other MTAs, if they quote this password when requesting the association.
NBS Routing Class	The National Bureau of Standards (NBS) Routing Class, determines the level of routing knowledge, in the form of O/R address components, held at an MTA. Routing Classes can be 1, 2, or 3. Class 1 corresponds to domain knowledge only (Country, ADMD, and PRMD).

Class 2 also includes knowledge about organizations and organization units, and class 3 about personal names.

The default is class 3.

Note

If you specify a class of less than 3, user addresses may be ambiguous.

For details about NBS routing classes, see the NBS implementor's agreement for OSI protocols (Stable Edition).

Network Address

The network address of the MTA. Specify the address in one of the following formats:

- *Nodename*
- A maximum of 15 digits preceded by a colon - X121 address.

The default is the MTA name (nodename).

Transport Protocol Selector

Selects the transport protocol to be used by this MTA.

The attribute must be entered as 2 hexadecimal digits per byte. For example, if the application uses printable characters, and you agree on the 2-byte code that corresponds to ASCII 17, then you must enter 3137. The maximum size is 32 bytes.

The default is 3432.

Checkpoint Size

A four-digit number representing the maximum number of 1024-byte data units, that can be transferred between adjacent MTAs before validation is required.

The minimum is zero, the default is 1.

Window Size

A four-digit number representing the maximum number of 1024-byte data units, that are allowed to be outstanding during data exchange over an active association.

The minimum is 1, and the default is 3.

Maximum transmit size

The maximum permitted data unit size in transmit mode.

Maximum receive size

The maximum permitted data unit size in receive mode.

Note

The Reliable Transfer Service (RTS) and Session Protocol Data Unit (SPDU) parameters represent initial negotiating values. Their runtime values are determined by negotiation between MTAs.

Non-default Associations

If you select option 4, Non-default Associations, on the Configure Local MTA menu (Figure 2-11), the Define Non-default Associations form is displayed, as in Figure 2-17.

```
) Define Non-default Associations (

Local MTA:

Adjacent MTAs:




Press:  <ADD>       to add a new adjacent MTA.
        <REMOVE>    to remove adjacent MTA.
        <TRANSMIT>  to modify association.
        <EXIT>      to return to previous menu.
```

FIGURE 2-17. Define Non-default Associations Form

You can select an association to an adjacent MTA to modify, add, or remove associations, using the TRANSMIT, ADD, and REMOVE keys.

When you select an association to modify, or add a new one, the Configure Associations form is displayed, as in Figure 2-18.

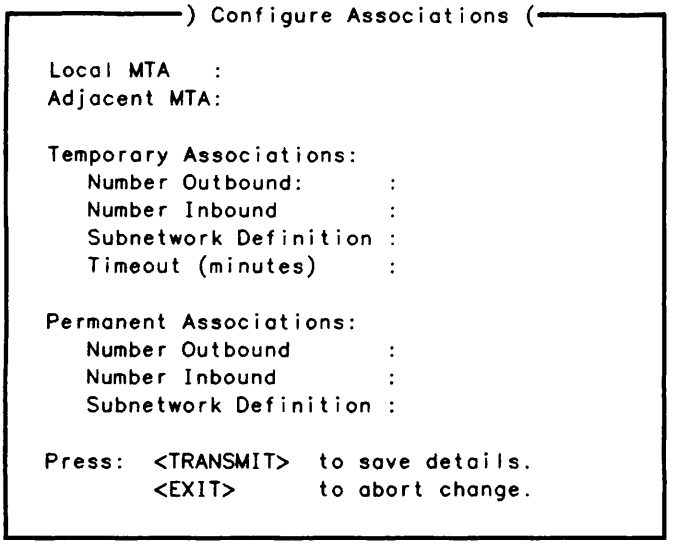

FIGURE 2-18. Configure Associations Form

To configure non-default associations, specify the numbers of inbound and outbound associations, the type of subnetwork, and, for temporary associations, the inactivity timeout period.

Outbound associations are those that carry data from the local MTA to the adjacent MTA. *Inbound* associations are those that carry data from the adjacent to the local MTA.

For further information about associations and subnetwork definitions, see earlier in this chapter.

CONFIG_X400 Option 4 - Configure Local Users

The CONFIG_X400 Main Menu option `Configure Local Users` (Figure 2-3), allows you to specify the O/R addresses and MTA attachments of users in your configuration.

When you select the `Configure Local Users` option, the Select Local User form is displayed, as in Figure 2-19.

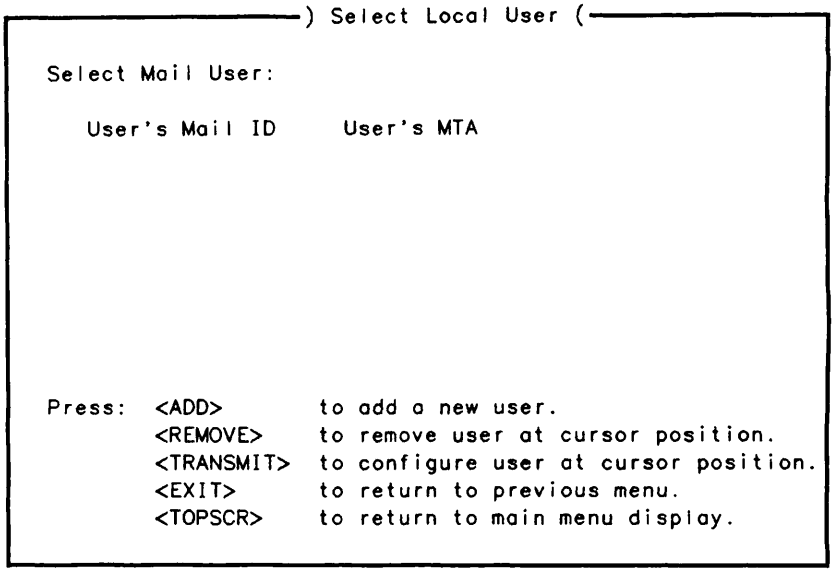

FIGURE 2-19. Select Local User Form

When you have selected a user to modify, the Configure User form pages 1, 2, 3, and 4 are displayed in sequence. For details see Figures 2-14 and 2-15.

CONFIG_X400 Option 5 - Define Remote MTAs

The CONFIG_X400 Main Menu option `Define Remote MTAs` (Figure 2-3), allows you to specify the protocols for communicating with remote MTAs, and the O/R addresses of users that are attached to them.

Data that you specify when defining remote MTAs is as follows:

- The remote domain
- MTA protocols and passwords
- O/R address spaces on specific MTAs

Remote MTAs are the MTAs, controlled by other administrators, with which you wish to communicate. They form the access points through which users in your configuration exchange messages with users in other configurations.

Remote MTAs are the responsibility of other administrators, and you may need to cooperate to obtain MTA configuration attributes such as name, password, protocol selectors and network address.

When you select the CONFIG_X400 main option Define Remote MTAs, the Select Remote MTA form is displayed, as in Figure 2-20.

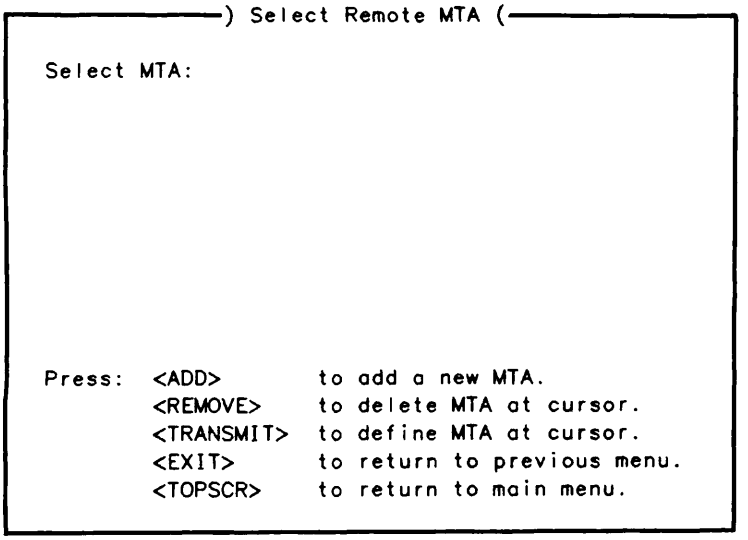

FIGURE 2-20. Select Remote MTA Form

The select remote MTA form allows you to select a remote MTA to change, and to add new remote MTAs to the configuration.

When you have selected an MTA to modify, or added a new one, the Define Remote MTA menu is displayed, as in Figure 2-21.

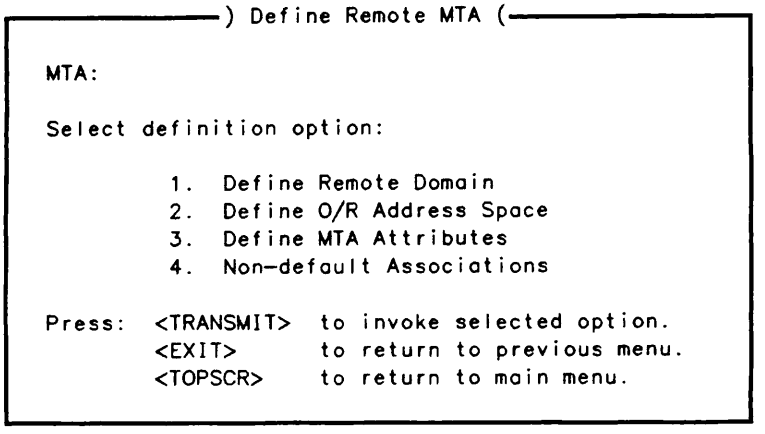

FIGURE 2-21. Define Remote MTA Menu

Define Remote Domain

If you select option 1, `Define Remote Domain`, on the Define Remote MTA menu (Figure 2-21), the Define Remote Domain form is displayed, as in Figure 2-22.

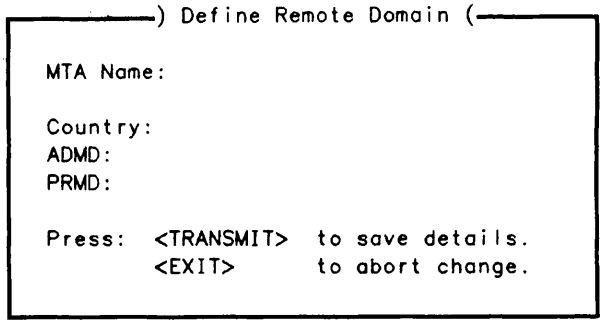

FIGURE 2-22. Define Remote Domain Form

Define the domain for a particular remote MTA by specifying Country, ADMD, and PRMD. Details of how to complete these fields can be found in the section CONFIG_X400 Option 1, `Set Domain Name`, earlier in this chapter.

Define O/R Address Space

If you select option 2, `Define O/R Address Space`, on the Define Remote MTA menu (Figure 2-21), the Select Remote MTA O/R Address Space form is displayed, as in Figure 2-23.

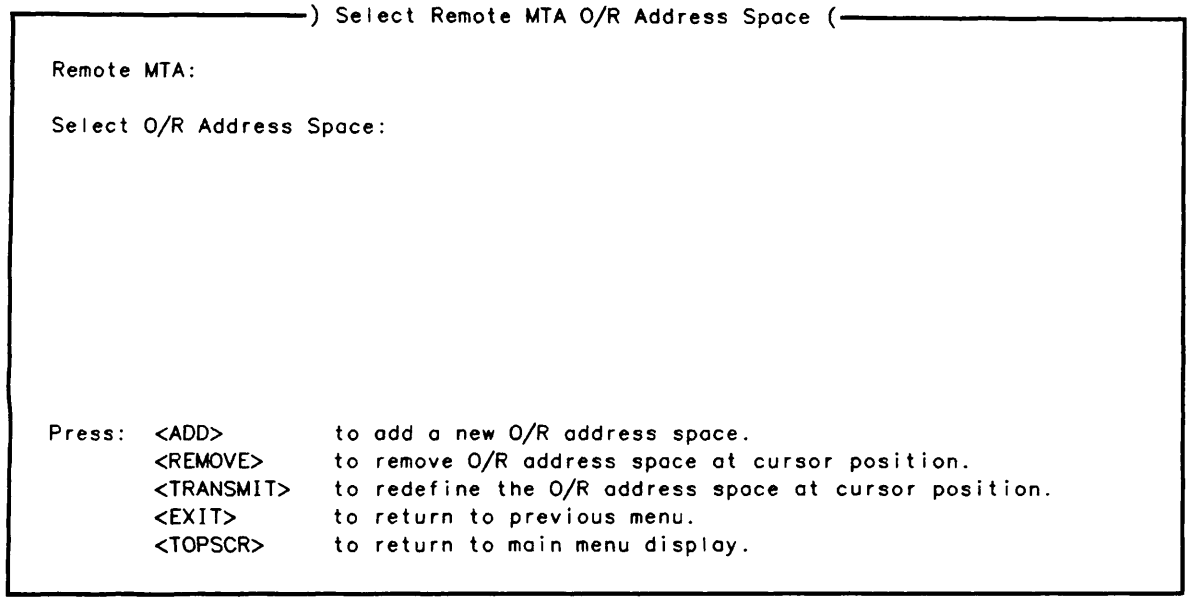

FIGURE 2-23. Select Remote MTA O/R Address Space Form

When you select a remote O/R address to change, the Define Remote O/R Address Space form pages 1 and 2 are displayed in sequence, as in Figure 2-24.

```
) Define Remote O/R Address Space (—) Page 1 of 2 (

MTA:

O/R Address Space:
   Country:      ADMD:                    PRMD:

   Organization Name:
     Organization:
     Units:

    Personal Name:
     Surname :                          Given Name:
     Initials:              Generation Qualifier:

Press:  <TRANSMIT>  to save user details.
        <EXIT>      to abort changes.
        <NXTPAGE>   to view next page.
```

```
) Define Remote O/R Address Space (—) Page 2 of 2 (

 O/R Address Space (cont.):

   Domain Defined Attributes:

      Type       Value



 Press:  <TRANSMIT>  to save O/R address details.
         <EXIT>      to abort changes.
         <PRVPAGE>   to view previous page.
```

FIGURE 2-24. Define Remote O/R Address Space Form, Pages 1 & 2

Define the remote O/R address space by completing any or all of the O/R address component fields.

O/R address components and their meanings are described under CONFIG_X400 Option 3, Configure Local MTAs, earlier in this chapter.

Define MTA Attributes

If you select option 3, Define MTA Attributes, on the Define Remote MTA menu (Figure 2-21), the Define Remote MTA Attributes form is displayed, as in Figure 2-25.

```
) Define Remote MTA Attributes (

MTA Name:
MTA's Password:                    Our Password:

Protocol Type:

Network Address            :
Transport Protocol Selector:

Press:  <TRANSMIT>  to save changes.
        <EXIT>      to abort changes.
```

FIGURE 2-25. Define Remote MTA Attributes form

Remote MTA attributes are described in the following list.

Attribute	*Description*
MTA Name	The name you must use to communicate with the remote MTA. Obtain the name from the MTA's administrator.
MTA's Password	The password that identifies the remote MTA. Obtain the password from the remote MTAs' administrator.
Our Password	The password that the remote MTA must supply in order to communicate with the local MTA. This password is assigned locally.
Protocol Type	The protocol type used by the remote MTA. The protocol type is set to either MOTIS or CCITT-84. The default is CCITT-84.
Network Address	The ISO network address (NSAP) of the MTA. Specify the address in one of the following formats: • Nodename • 15 digits preceded by a colon, this is the X121 address.

If in doubt, allow the default to be selected.

Transport Protocol Selector	Selects the ISO transport protocol. Obtain from the MTAs administrator. Enter as 2 hexadecimal digits per byte. For example, if the application uses printable characters, and you agree

on the 2-byte code that corresponds to ASCII 17, then you must enter 3137.

The maximum permitted size is 32 bytes, and the default value is 3432.

Non-default Associations

If you select option 4, Non-default Associations, on the Define Remote MTA menu (Figure 2-21), the Define Non-default Associations form is displayed, as in Figure 2-26.

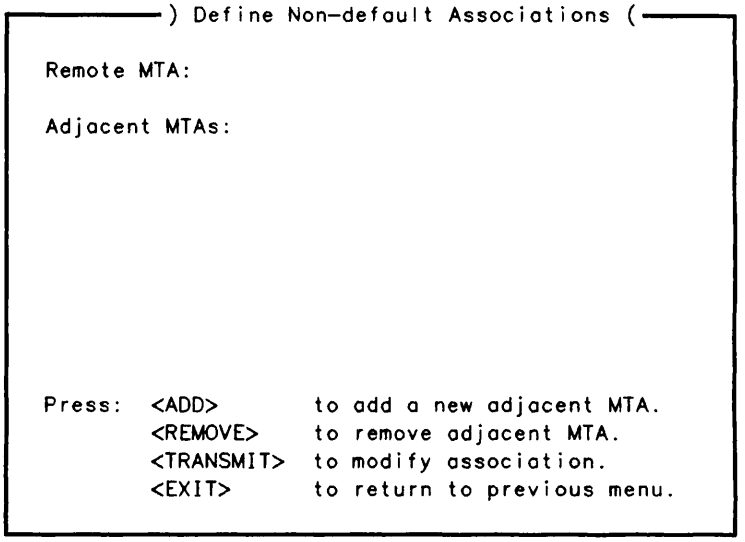

FIGURE 2-26. Define Non-default Associations Form

You can select an association to an adjacent MTA to modify, add, or remove associations, using the TRANSMIT, ADD, and REMOVE keys.

When you select an association to modify, or add a new one, the Configure Associations form is displayed, as in Figure 2-27.

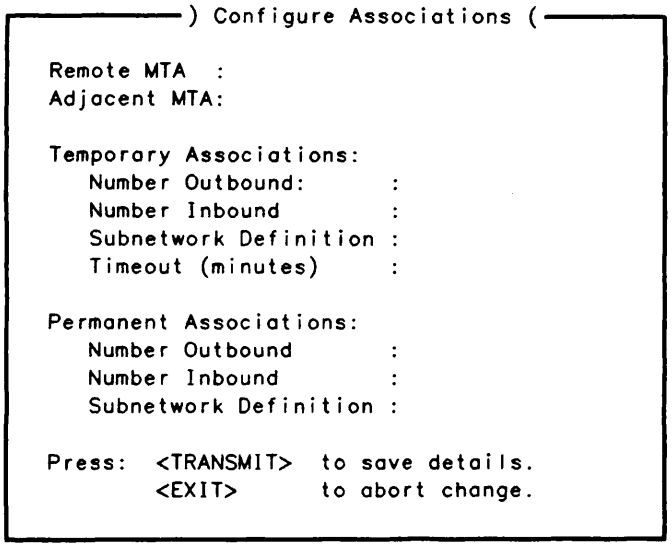

FIGURE 2-27. Configure Associations Form

To configure non-default associations, specify the numbers of inbound and outbound associations, the type of subnetwork, and, for temporary associations, the inactivity timeout period.

Outbound associations are those that carry data from the local MTA to the adjacent MTA. *Inbound* associations are those that carry data from the adjacent to the local MTA.

For further information about associations and subnetwork definitions, see earlier in this chapter.

CONFIG_X400 Option 6 - Define Remote O/R Addresses

The CONFIG_X400 Main Menu option `Define Remote O/R Addresses` (Figure 2-3), allows you to specify the O/R address space(s) associated with a specific remote MTA. It allows you to define a new address space, or modify an existing one.

You select the address space to modify, or define a new one, on the Select Remote O/R Address form, which is illustrated in Figure 2-28.

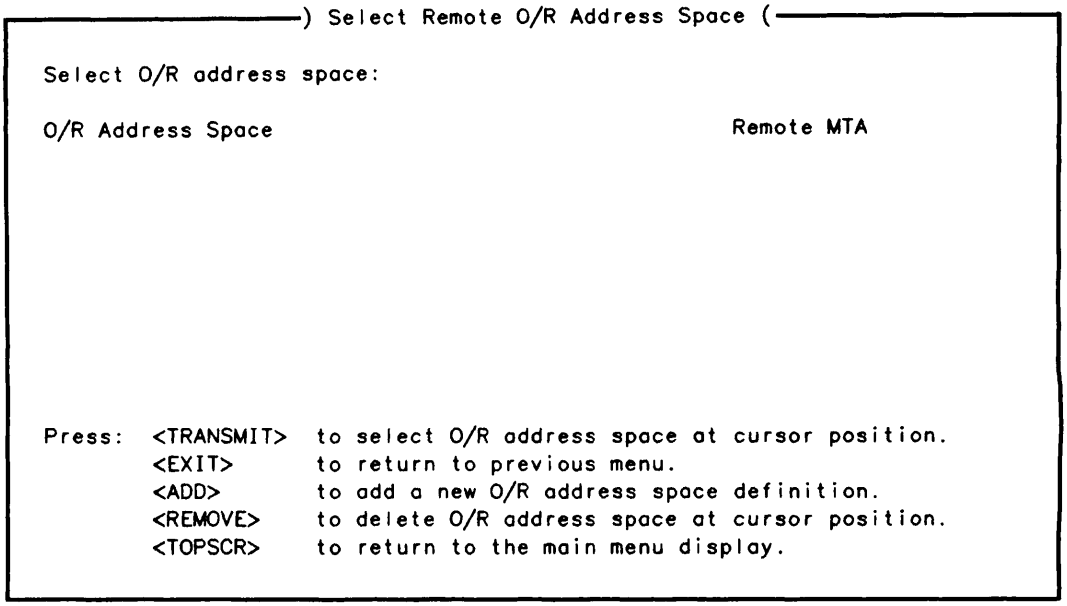

FIGURE 2-28. Select Remote O/R Address Space Form

When you have selected an address space, the Define Remote O/R Address Space form pages 1 and 2 are displayed in sequence, as in Figure 2-24, earlier in this chapter.

CONFIG_X400 Option 7 - Verify Configuration

The CONFIG_X400 Main Menu option `Verify Configuration` (Figure 2-3), allows you to verify your configuration.

While verification is taking place, the following message appears on screen.

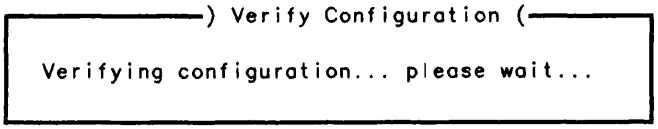

FIGURE 2-29. Verify Configuration Display

Errors that can be reported are:

- Alternate Recipient Mail ID is not known.
- Alternate Recipient MTA is not known.
- Local Users O/R Address attributes do not match the domain name.
- Missing NSAP for an MTA.
- Undefined Subnetwork Definition referenced.
- Remote MTA has no associations to local MTAs.

If the configuration is not verified, it is invalid and cannot be used to start Prime X.400 on the system. The administrator should update the configuration to correct the errors, and then select the `Verify Configuration` option in the Main Menu, (Figure 2-3), once more.

CONFIG_X400 Option 8 - Save Configuration

The option `Save Configuration` in the CONFIG_X400 Main Menu, (Figure 2-3), allows you to save your configuration to disk file.

You can use the Save function at any time, either to store an unfinished configuration for later modification, or to store a completed and verified configuration permanently to disk.

When you select the `Save Configuration` option on the CONFIG_X400 Main Menu (Figure 2-3), the Save Configuration form is displayed.

```
) Save Configuration (

Configuration File:

Comment:

Press:  <TRANSMIT>  to save configuration.
        <EXIT>      to abort save and return to menu.
```

FIGURE 2-30. Save Configuration Form

Configurations are automatically verified before being saved to disk (specific errors are not reported). If the check fails, you are queried about the save. If the check succeeds, the configuration is saved to disk.

Configuration File can be any PRIMOS pathname.

You can add your own remarks on the `Comment:` line. The comment is displayed in the configuration file information display when you invoke CONFIG_X400, as in Figure 2-1.

CONFIG_X400 Option 9 - List Configuration

The CONFIG_X400 Main Menu option `List Configuration` (Figure 2-3) allows you to save a listing of your configuration to disk file.

When you select the List Configuration option, the List Configuration form is displayed.

```
) List Configuration (

Listing filename:

      Overwrite:
         Append:

Press:  <TRANSMIT>  to start listing to file.
        <EXIT>      to return to main menu.
```

FIGURE 2-31. List Configuration Form

Specify any filename.

To overwrite, or append to, an existing file, type YES or Y after the appropriate prompt.

3

OPERATION AND MONITORING

Introduction

This chapter describes the ADMIN_X400 command, its options and subcommands, that you use to start, stop, monitor, and control Prime X.400 on the system.

The ADMIN_X400 Command

The ADMIN_X400 command allows you to control and monitor Prime X.400. You can use the command in two ways.

- Using command line options you can start and stop Prime X.400 on the system.
- Through the subcommand environment that you enter by invoking the command without options, you can
    - Display users' status and O/R addresses
    - Display MTA configuration data
    - Display queues
    - Display all errors at your terminal

User Access

ADMIN_X400 is an operator command and is normally invoked at the supervisor terminal. Options -START and -STOP are restricted to the supervisor terminal only.

Users access to the display and monitoring subcommands is enabled using the access category ADMIN.ACAT in the PRIME_X400* directory. Users with at least U (Use) access assigned in this category can invoke the subcommands from a normal terminal. (See the *PRIMOS Commands Reference Guide* for access rights).

Command Syntax

The syntax of the ADMIN_X400 command is as follows:

▶ **ADMIN_X400 [options]**

Options are described in the following list.

Option	*Description*
-START [MTA name] [-CONFIG filename]	Starts a specific Prime X.400 MTA on this node. If you do not give an MTA name, the local node name is used. To start the MTA with a specific configuration, specify -CONFIG followed by a filename. If you do not specify a filename, the default configuration file, pathname PRIME_X400*>PRIME_X400.CONFIG, is used. If the startup configuration file is corrupt, Prime X.400 starts with its existing routing tables.

-STOP [-FORCE] Shuts down Prime X.400 on the system. To forcibly stop the server, specify -FORCE.

-ON nodename Invokes the subcommand environment for the MTA on the specified node. This option can not be used with the -START and -STOP options.

{ **-HELP** | **-H** } [{ **-NO_WAIT** | **-NW** }]

Explains how to use the command. This option cancels any other options on the command line. If you specify -NO_WAIT, the display is not paginated at your terminal. The same information is available through the PRIMOS HELP subsystem.

-USAGE Gives you the command syntax in brief. This option cancels all others on the command line.

Starting and Stopping Prime X.400

To start Prime X.400 on the system, type the following command at the supervisor terminal:

```
admin_x400 -start
```

This starts Prime X.400 with the default configuration. To use a specific configuration, specify the -CONFIG filename option.

When user X400_SERVER appears on the STATUS USERS list, the Prime X.400 subsystem is up and running. It may, however take a few minutes for it to reach the state whereby mail can be processed, or the ADMIN_X400 subcommand environment invoked. The ADMIN_X400 -START command can be included in your PRIMOS.COMI system startup file if you wish.

To stop Prime X.400, type:

```
admin_x400 -stop
```

This shuts down all MTA associations and logs the server out, after ensuring that all active sessions are closed.

Display and Control Subcommands

This section gives a general description of the Display and Control Subcommands: detailed descriptions are given in the subsequent sections. You invoke the display and control subcommands from a special environment that you enter by invoking the ADMIN_X400 command with no options. For example:

```
OK, admin_x400

[ADMIN_X400 Rev. 1.0.0-21.0.3 Copyright (c) 1988, Prime Computer, Inc.]

Establishing connection to X400... please wait

Welcome.
Console status : Operator present.
Product : P-X400      Type : OPER   State : ATTACHED   Bcast : Y
Console status : Current mode P-X400.
```

The system is then ready to receive subcommands.

To quit from ADMIN_X400 and return to PRIMOS, type Q or QUIT.

Display and Control Subcommands

Subcommands are of two types:

- Display commands
- Control commands

Display commands give you up-to-the-minute status information about MTAs and users in the configuration. Display commands are:

- DISPLAY-USER
- DISPLAY-MTA
- DISPLAY-ASSOC
- DISPLAY-QUEUE
- DISPLAY-ORNAME

Control commands allow you to control the operation of Prime X.400 on the system. The Control command is:

- X400-LOGGING

Help

To obtain help on subcommand syntax and usage, type HELP within the subcommand environment.

Syntax

Subcommands consist of a hyphenated verb, usually followed by an operand. The operand identifies the entity on which the command operates; for example, in display commands it identifies the entity to display. Subcommands are typed in upper or lower case.

Operands for subcommands consist of qualified keywords that identify a specific entity or group of entities. Keyword qualifiers are either further keywords linked to the entity keyword by the (=) character, or Prime X.400 routing table identifiers enclosed in parentheses. Routing table identifiers are Mail IDs or MTA names.

The following are examples of valid operands:

```
USER=DEFAULT
USER(JOE)
```

The following are examples of full subcommand syntax:

```
DISPLAY-USER USER(JOE)

DISPLAY-USER
```

Command Line Editing

Erase and kill characters are those that are already set as part of your PRIMOS environment.

Display Subcommands

This section describes ADMIN-X400 display subcommands, their syntax and hierarchy, and gives examples of their use. Using these commands you can display

- The X.400 communication status of users
- Adjacent MTAs
- Details of MTA Associations
- Gateway connections
- Users' full O/R addresses
- Status of user and MTA send/receive queues.

Display subcommands and their functions are described in the following list.

Subcommand	*Function*
DISPLAY-USER	Displays the X.400 communication status of local users.

DISPLAY-MTA	Displays the names of adjacent MTAs, and the numbers of associations they have with the local MTA.
DISPLAY-ASSOC	Displays the status of associations with adjacent MTAs. Data returned includes ownership, type, and current status of the connection.
DISPLAY-QUEUE	Displays details of send/receive queues for both users and MTAs.
DISPLAY-ORNAME	Displays the full O/R address of a specific user.

Primary and Secondary Commands

Display subcommands are of two types: primary and secondary.

Primary commands give direct information about the main configuration entities, that is, users, MTAs, and gateways. DISPLAY-USER and DISPLAY-MTA are primary commands.

Secondary commands give specific information about the main configuration entities, and require the prior specification of a user, MTA, or gateway. DISPLAY-ORNAME, DISPLAY-QUEUE and DISPLAY-ASSOC are secondary commands.

The requirements of the secondary display commands are illustrated in Figure 3-1.

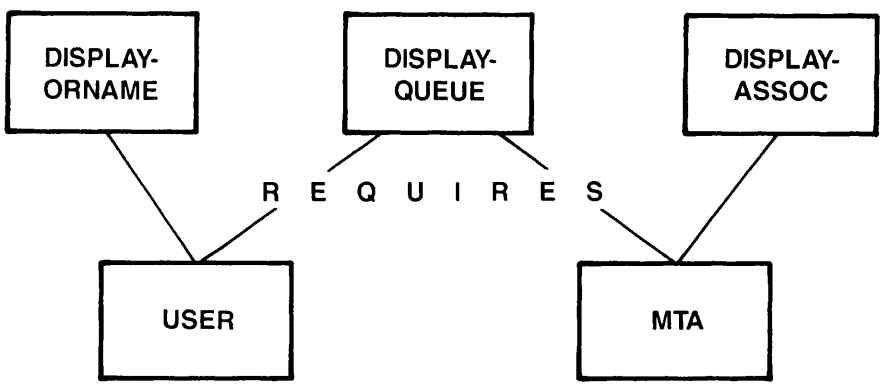

FIGURE 3-1. Hierarchy of Display Commands

Display Defaults

ADMIN_X400 stores the last primary and the last secondary display argument that you specify. These become the defaults for all subsequent commands until you change them.

For example, if you use the `DISPLAY-USER USER=JOE` command to display the status of mail user JOE, the user name JOE is stored as the default display argument. If you now use the DISPLAY-ORNAME command with the default specifier, the command displays the O/R address of user JOE.

Secondary commands require a primary argument. Specify either the default entity using the DEFAULT keyword, or one of your choice. The following sections give details of the display commands, their syntax, operands, and codes, together with examples of displays.

▶ **DISPLAY-USER** **USER=DEFAULT**
USER(Mail ID)
USER=ALL

The DISPLAY-USER command displays the communication status of Prime X.400 users. You can display information about specific users, the default user, or all users.

Specify individual users by the *Mail ID* assigned in CONFIG_X400. To display all users, use the DISPLAY-USER USER=ALL command.

Operands

Operands and their meaning are described below.

Operand	*Meaning*
USER=DEFAULT	Selects the default user.
USER(Mail ID)	Specifies a particular user by the Mail ID. To display all users in the configuration, use the USER=ALL operand. This operand updates the default user.
USER=ALL	Selects all users.

Status Codes

Communication status codes indicate the current state of the user's X.400 session. Their meanings are described in the following list.

Code	*Meaning*
RESET	User not logged in
IDLE	User logged in, inactive
PND RSP	Awaiting reply from user

PND CNF User awaiting confirmation from system

PND ACK System awaiting acknowledgement from user

PND CLS Logoff (from Prime X.400) in progress

Example Displays

```
display-user user=all
USER                STATUS
BILL                IDLE
JOE                 IDLE
End-Of-Table

display-user user(bill)
Application bill not found.

display-user user(BILL)

USER                STATUS
BILL                IDLE
End-Of-Table

display_user user=default
USER                STATUS
BILL                IDLE
End-Of-Table
```

Messages From DISPLAY-USER

`User <Mail ID> not found.`

The user you specified does not exist in the configuration.

▶ **DISPLAY-MTA** **MTA=DEFAULT**
MTA(MTA name)
MTA=ALL

The DISPLAY-MTA command displays the number of associations configured from adjacent MTAs to your local MTA. Specify MTAs by *MTA name*, DEFAULT, or ALL.

Operands

Operands and their meanings are described below.

Operand	*Meaning*
MTA=DEFAULT	Selects the default MTA.
MTA(MTA name)	Selects a specific adjacent MTA. This operand updates the default MTA.

MTA=ALL Selects all adjacent MTAs.

Example Displays

```
display-mta mta=all
ADJACENT MTA        ASSOCIATIONS
NODE1               0002
NODE2               0002
End-Of-Table

display-mta mta(NODE1)
ADJACENT MTA        ASSOCIATIONS
NODE1               0002
End-Of-Table

display-mta mta(NODDY)
Adjacent mta NODDY not found.
```

Messages From DISPLAY-MTA

`Adjacent mta <MTA name> not found.`

The MTA specified does not exist in the configuration.

▶ DISPLAY-ASSOC MTA=DEFAULT MTA(MTA name)

The DISPLAY-ASSOC command displays details about associations configured to a specific adjacent MTA. Specify the adjacent MTA by its *MTA name*, or select the command default.

The command displays the following information:

- Name of the associated MTA
- Association ownership (local, permanent/temporary, remote)
- Association type (monolog, or dialog)
- Association state (open, closed, opening, closing, sending, receiving, error, or aborted)
- Queue entry identifier for the message being sent

Operands

Operands and their meanings are described below.

Operand	*Meaning*
MTA=DEFAULT	Selects the default MTA.
MTA(MTA name)	Selects a specific MTA by its *MTA name*. This operand updates the default MTA entry.

Association Ownership

Association ownership is the capacity to control message transfer between the two associated MTAs.

There are three possible ownership states. These are described as follows:

State	*Meaning*
TEMP	Locally owned and temporary
PERM	Locally owned and permanent
REMOTE	Remotely owned

Association Types

Association type relates to the duplex nature (half or full) of the association between two adjacent MTAs.

There are two possible association types:

Type	*Meaning*
MONOLOG	Data flow allowed in one direction only
DIALOG	Data flow allowed in both directions

Association States

Association state is the current, runtime state of the association.

There are eight possible association states. These are described in the following list:

State	*Meaning*
OPEN	The association is open and idle.
CLOSED	The association is closed.
OPENING	The association is in the process of being opened.
CLOSING	The association is in the process of being closed.
SENDING	A message is being sent on the association.
RECEIVING	A message is being received on the association.
ERROR	The association cannot be opened.
ABORTED	The association has been aborted.

Example Displays

```
display-assoc mta=default
Adjacent_mta : NODE1
NAME            OWNER     TYPE      STATE     SND-ID
ASSOC1          REMOTE    MONOLOG   CLOSED    0000 00
ASSOC2          TEMP      MONOLOG   CLOSED    0000 00
End-Of-Table
```

▶ DISPLAY-ORNAME USER=DEFAULT
USER(Mail ID)

The DISPLAY-ORNAME command displays the full O/R address of a specific user, in a tabulated format. Specify the user by the configured Mail ID, or select the default. To determine Mail IDs for users, use the DISPLAY-USER USER=ALL command.

Operands

Operands and their meanings are described as follows:

Operand	*Meaning*
USER=DEFAULT	Selects the default user.
USER(Mail ID)	Selects a specific user by the *Mail ID.* This operand updates the default user.

O/R Address Components

There are eleven possible address components. These are described in the following list:

Component	*Meaning*
CTY	Country Name. This is assigned by the X.400 regulatory authorities.
ADM	Administration Domain Name. This is assigned by the X.400 regulatory authorities.
PDM	Private Domain Name.
ORN	Organization Name.
ORU	Organization Unit.
SUR	Surname.
GIV	Given Name.
INI	Initials.

GEN Generation Qualifier.

DDT Domain Defined Attribute Type.

DDV Domain Defined Attribute Value.

Example Displays

```
display-orname user(BILL)
User : BILL
O/R NAME
CTY : UK
ADM : MHS
PDM : PRIME
ORN : DC
ORU : OSI
SUR : SHAKESPEARE
GIV : WILLIAM
End-Of-Table
```

Messages From DISPLAY-ORNAME

`User <Mail ID> not found.`

The user you specified does not exist in the configuration.

▶ **DISPLAY-QUEUE** **USER=DEFAULT** **USER(Mail ID)** **MTA=DEFAULT** **MTA(MTA name)** **QUEUE=DEFAULT** **QUEUE=IN** **QUEUE=OUT** **QUAL=SUMMARY** **QUAL=FULL**

The DISPLAY-QUEUE command lists the status of send and receive message queues (IN and OUT queues) for users and adjacent MTAs. You can select a summary, or a detailed display.

The *summary* display contains the following information:

- Queue type (IN/OUT)
- Total number of messages on the queue
- Numbers of high, medium, and low priority (URGENT, NORMAL, NON-URG) messages

The *full* display lists the summary information plus details of each message:

- A serial number that identifies the message uniquely
- The date and time the message was added to the queue
- The message type (IM-UAPDU, SR-UAPDU, DR-MPDU, PR-MPDU)
- Message priority (NORMAL, URGENT, NON-URG) messages

Operands

Operand	*Description*
USER=DEFAULT	Selects the default user.
USER(Mail ID)	Selects a specific user by the user's *Mail ID*. This option resets the default user.
MTA=DEFAULT	Selects the display default adjacent MTA.
MTA(MTA name)	Selects a specific adjacent MTA. This option resets the default MTA.
QUEUE=DEFAULT	Selects the existing default queue type.
QUEUE=IN	Selects the IN or *receive* queue. This option resets the display default queue type.
QUEUE=OUT	Selects the OUT or *send* queue. This option resets the display default queue type.
QUAL=SUMMARY	Selects the summary display.
QUAL=FULL	Selects the detailed display.

Message Types

There are four possible message types.

Type	*Description*
IM-UAPDU	Interpersonal Message in a User Agent Protocol Data Unit.
SR-UAPDU	Status Report in a User Agent Protocol Data Unit. This is user acknowledgement to the system.
DR-MPDU	Delivery Report in a Message Protocol Data Unit. This is system acknowledgement to the user.
PR-MPDU	Probe Report in a Message Protocol Data Unit. This is system message to verify an MHS route.

For further information about message types, see the CCITT X.400 series documentation.

Issue State

A message can be on, or off, issue. On issue means that the message is being actively processed by Prime X.400.

Example Displays

```
display-queue queue=out qual=summary
User : BILL
Queue: OUT Tot: 0000 High: 0000 Normal: 0000 Low: 0000

display-queue queue=in qual=full
User : BILL
Queue : IN Tot: 04 High: 01 Normal: 02 Low: 01
MSG-ID  DATE        TIME        MSG-TYPE   PRIORITY  ISSUE
There are no messages on this queue at this time.
End-Of-Table
```

Control Subcommand

This section describes the ADMIN_400 control subcommand, and gives an example of its use. Using the command you can

- Control the display of error messages at the operator's terminal

The command and its function is described below.

Command *Function* **X400-LOGGING**
Enables/disables error display to the operator's terminal

Descriptions of the command and its syntax follow.

```
                    STATUS
▶ X400-LOGGING      STATUS=ON
                    STATUS=OFF
```

The X400-LOGGING command enables and disables full error logging at the user's terminal. It is also used to display logging status.

Error messages from Prime X.400 are normally logged only in the Prime X.400 journal (como) log. When error logging is enabled, error messages are also echoed to the terminal.

There are three possible operands to the command. These are described as follows:

Operand	*Description*
STATUS	Displays current logging status (enabled or disabled)
STATUS=ON	Enables error logging to the user's terminal

STATUS=OFF Disables error logging to the user's terminal

PART II
EXAMPLES

4

CONFIG_X400 EXAMPLES

Introduction

This chapter details the step-by-step procedure of how to set configuration defaults, configure a single local MTA, add other local MTAs, configure a remote MTA, and configure large numbers of local users. Each sequence is illustrated with examples of the screen dialogue.

Refer throughout this chapter to Chapter 2, PRIME X.400 CONFIGURATION for further details.

EXAMPLE I - Setting Configuration Defaults

This example shows you how to Set Configuration Defaults.

Stages in Setting Configuration Defaults

1. Invoke CONFIG_X400 Command.
2. Define Alternate Recipient.
3. Define Subnetwork Addresses.
4. Define MTA Associations.
5. Verify Configuration.
6. Save Configuration.
7. List Configuration.
8. Start Prime X.400.

Figure 4-1 illustrates the hierarchy of screen forms that you navigate to set configuration defaults.

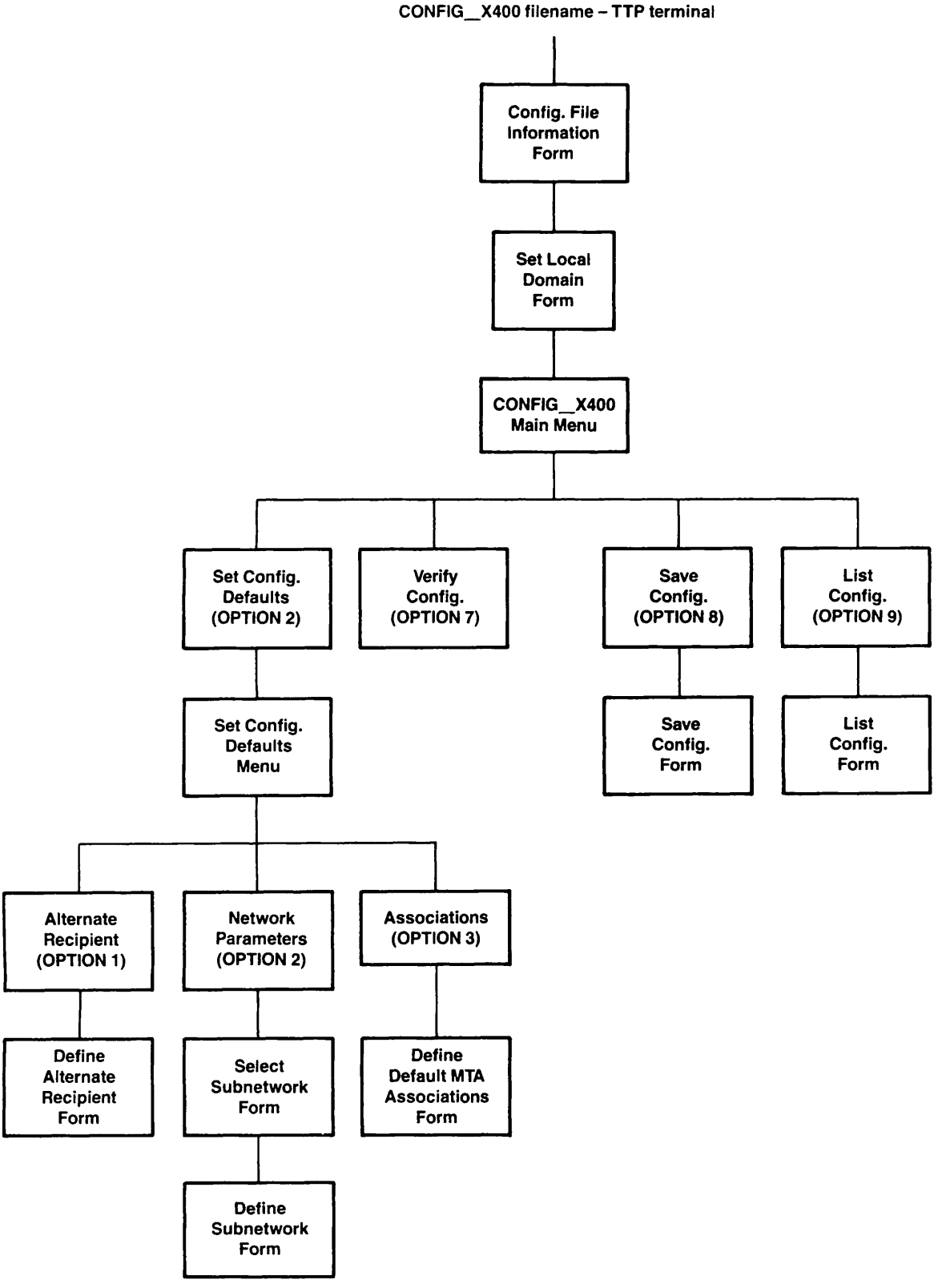

FIGURE 4-1. Hierarchy to Set Configuration Defaults

Stage 1 - Invoke CONFIG_X400 Command

To start configuring, type the following command at the supervisor terminal, then press TRANSMIT:

```
CONFIG_X400 filename -TTP terminal
```

where

filename — Allows you to create a new configuration. If the file does not already exist, a new file is created to contain the configuration. If no filename is given, then the default configuration file is used.

terminal — Specifies the terminal type you are using. Any terminal defined in the Priforma database is valid. If you have a .TERMINAL_TYPE$ global variable defined, then this option can be omitted.

Information about the input configuration file is then displayed as in Figure 4-2.

```
Config File: filename.CONFIG

Revision Number:
Last Updated   :
Updated by user:
Updated on node:

Comment:
```

FIGURE 4-2. Configuration File Information Form

If you entered a new file name when invoking the CONFIG_X400 command, an Operator prompt is displayed `Config file does not exist. Do you wish to create a new configuration? (y/n):`, type Y and press TRANSMIT. You are presented with a form to set the local domain; this is the same form that is obtained with option 1, `Set Local Domain`, of the CONFIG_X400 Main Menu (Figure 2-3 and Figure 2-4). Figure 4-3 illustrates the Set Local Domain Form.

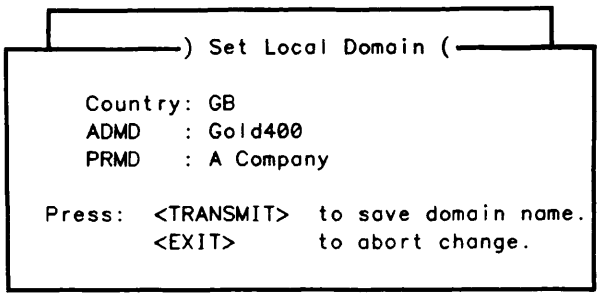

FIGURE 4-3. Set Local Domain Form

You must set the local domain by specifying Country, ADMD, and PRMD.

Field	*Description*
Country	The country in which the network operates. This is defined by international regulatory authorities. Specify either: • A 3-digit code as defined by CCITT X.121 (for example, the UK code is 234, the US code is 311). • A 2-letter code as defined by ISO 3166/ALPHA-2 (for example, GB, US). For further details about Country codes, refer to the relevant CCITT and ISO data.
ADMD	The main administration domain with which you are associated. If you are connecting to a public service mail system, use the name they give you. If you are connecting to a private mail application, choose one by bilateral agreement, or leave the field blank. The ADMD can contain a maximum of 16 characters.
PRMD	The private administration domain with which you wish to be associated. If you are associated with an ADMD, the PRMD may be defined by the administration of the ADMD. If you are connecting to an independent mail application, agree the name with the administrator responsible for the application. The PRMD can contain a maximum of 16 characters.

When you have set the local domain, press TRANSMIT to save the details. You are presented with the CONFIG_X400 Main Menu, as illustrated in Figure 4-4.

```
) CONFIG_X400 - Main Menu (

Select configuration option:

        1.  Set Local Domain
        2.  Set Configuration Defaults

        3.  Configure Local MTAs
        4.  Configure Local Users

        5.  Define Remote MTAs
        6.  Define Remote O/R addresses

        7.  Verify Configuration
        8.  Save Configuration
        9.  List Configuration

Press:  <TRANSMIT>  to invoke selected option.
        <EXIT>      to exit the configurator.
```

FIGURE 4-4. CONFIG_X400 Main Menu

Note

To select an option from the Main Menu, use the cursor keys to position the cursor at the required option and press TRANSMIT. Alternatively, enter the number of the required option and press TRANSMIT.

Stage 2 - Define Alternate Recipient

Select option 2, `Set Configuration Defaults`, from the CONFIG_X400 Main Menu (Figure 4-4), and press TRANSMIT.

You are presented with the Set Configuration Defaults Menu as illustrated in Figure 4-5.

```
) CONFIG_X400 - Main Menu (
) Set Configuration Defaults (

Select option:

            1.  Alternate Recipient
            2.  Network Parameters
            3.  Associations

Press:  <TRANSMIT>  to invoke selected option.
        <EXIT>      to return to main menu.
```

FIGURE 4-5. Set Configuration Defaults Menu

Select option 1, Alternate Recipient, from the Set Configuration Defaults Menu. You are presented with the Define Alternate Recipient Form as illustrated in Figure 4-6.

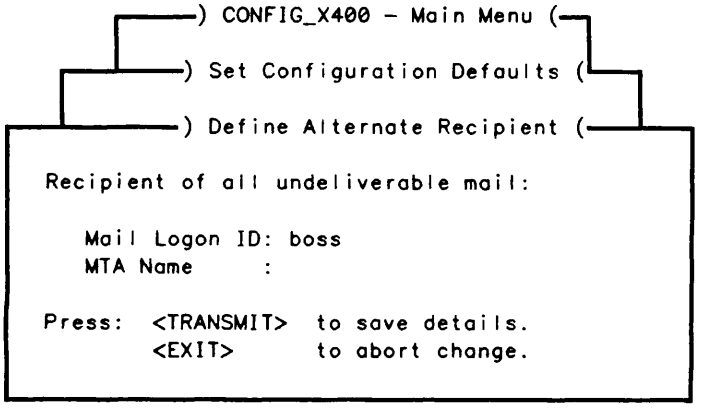

FIGURE 4-6. Define Alternate Recipient Form

Complete the form. The alternate recipient may be a user on one of the local MTAs, or on one of the remote MTAs (see EXAMPLE IV - Configuring a Remote MTA).

Stage 3 - Define Subnetwork Addresses

To define subnetwork addresses, select option 2, Network Parameters from the Set Configuration Defaults Menu (Figure 4-5). You are presented with the Select Subnetwork form as illustrated in Figure 4-7.

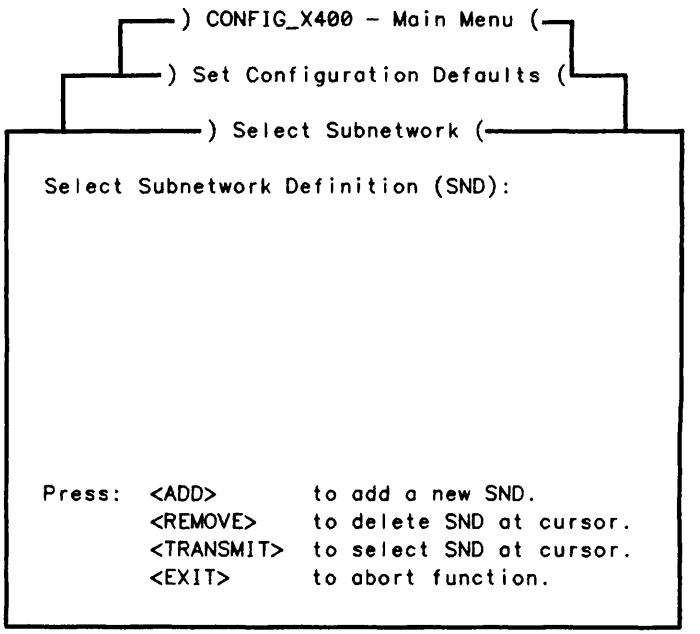

FIGURE 4-7. Select Subnetwork Form

Press ADD to define the subnetwork. Figure 4-8 illustrates the Define Subnetwork form.

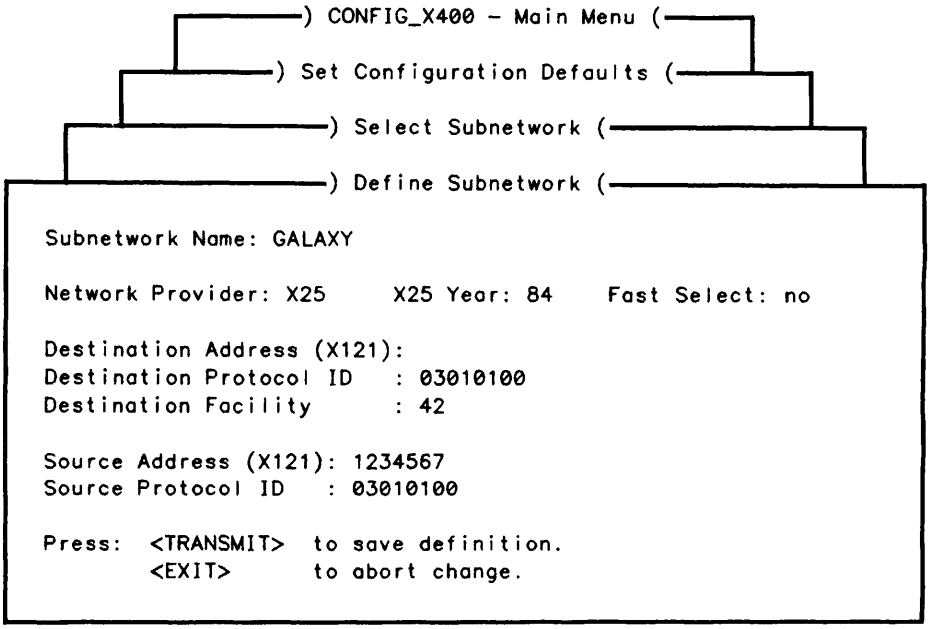

FIGURE 4-8. Define Subnetwork Form

The Define Subnetwork form is used to set up communication parameters between a local Prime MTA and another MTA (local or remote).

The Subnetwork definition fields are described in the following list.

Field	*Description*
Subnetwork Name	Your name for the subnetwork.
Network Provider	Always X25.
X25 Year	Enter the X25 year; for example, 1980.
Fast Select	Selects the X25 Fast Select facility. Enter YES or NO.
Destination Address (X121)	The X121 address on outgoing calls. By default, this is computed from the NSAP address.
Destination Protocol ID	The X25 protocol at the destination MTA. Four bytes, entered as ASCII-Hex.
Destination Facility	The X.25 Facilities to propose when making an X.25 call to the remote system which operates the Remote MTA. For details of how to encode X.25 Facilities, refer to the *Primenet Programmer's Guide.* Facilities are not normally required for correct operation.
Source Address (X121)	The X121 address for incoming calls. Enter as a full X121 address, or as an X121 subaddress in the form "+<digits>", which is concatenated with the PRIMENET address to form the full X.121 address. Source Address should be distinct from any subaddress used for PRIMENET Route-through.
Source Protocol ID	The X25 protocol at the source MTA. Enter four bytes of ASCII-Hex.

Complete the Define Subnetwork form and press TRANSMIT. The Select Subnetwork form is resumed, displaying the subnetwork just created.

Stage 4 - Define MTA Associations

An association is a connection (a logical link) to another MTA, whether it is Local or Remote. There are two types of association between MTAs; temporary and permanent.

A temporary association allows you to send messages to an MTA and has a timeout period, in cases of inactivity. The maximum number of associations for both local and remote MTAs is 999. The maximum timeout is 999 minutes.

A permanent association does not have a timeout period. Messages can be sent to an MTA and, if that MTA does not respond, the messages are held in the server until connection is resumed. The maximum number of associations for both local and remote MTAs is 999.

By default, all local nodes/MTAs are mutually interconnected with a single temporary association between each.

Select option 3, `Associations` from the Set Configuration Defaults Menu (Figure 4-5). Figure 4-9 illustrates the Define Default MTA Associations Form.

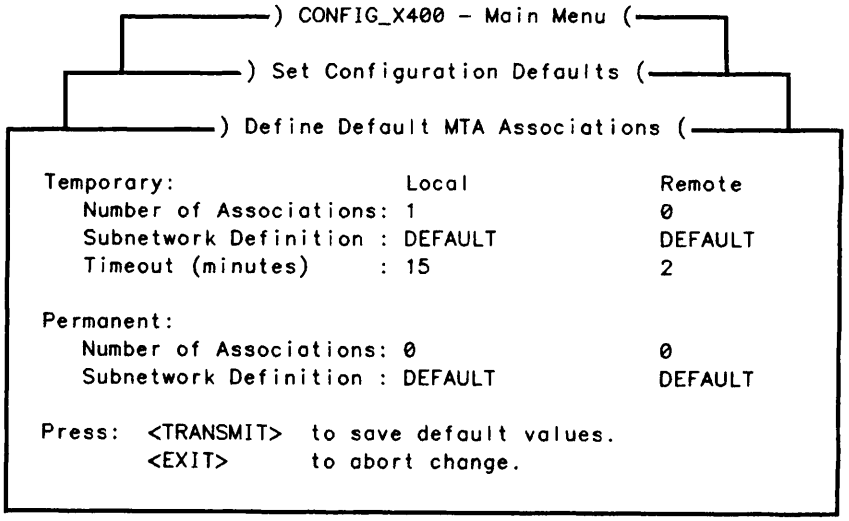

FIGURE 4-9. Define Default MTA Associations Form

Complete the details on the Define Default Associations Form by specifying the number of temporary and permanent associations required for local and remote MTAs and, the type of subnetwork involved.

For temporary associations only, specify the inactivity timeout period (temporary associations are only maintained while messages are being exchanged).

When you have completed the form, press TRANSMIT to save the details.

Stage 5 - Verify Configuration

To check that all information has been entered correctly, return to the CONFIG_X400 Main Menu (Figure 4-4) and select option 7, `Verify Configuration`.

While verification is taking place, the following message appears on screen.

```
) Verify Configuration (

Verifying configuration... please wait...
```

If the configuration fails the check, it is invalid and cannot be used to start Prime X.400 on the system.

Stage 6 - Save Configuration

Select option 8, Save Configuration from the CONFIG_X400 Main Menu (Figure 4-4). You are presented with the Save Configuration Form as illustrated in Figure 4-10.

```
) CONFIG_X400 - Main Menu (
) Save Configuration (

Configuration File: EXAMPLE.CONFIG

Comment: First configuration example

Press:  <TRANSMIT>  to save configuration.
        <EXIT>      to abort save.
```

FIGURE 4-10. Save Configuration Form

Complete the Save Configuration Form by entering a comment (if required) and pressing TRANSMIT. Pressing TRANSMIT prompts you to create the file. Enter Y and press TRANSMIT.

Stage 7 - List Configuration

A listing of the configuration file can be produced for spooling to a printer, using option 9, List Configuration from the CONFIG_X400 Main Menu (Figure 4-4).

Figure 4-11 illustrates the List Configuration Form.

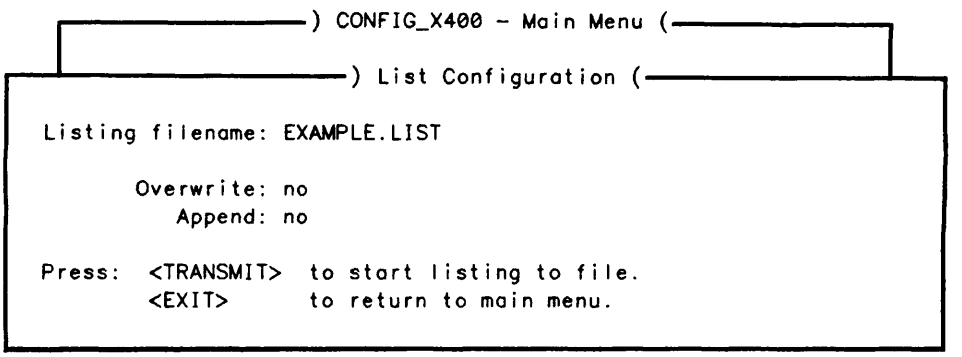

FIGURE 4-11. List Configuration Form

Stage 8 - Start PRIME_X400

When a new configuration has been created and saved, use it to start PRIME_X400 on the system by typing the following command at the supervisor terminal:

```
ADMIN_X400 -START
```

EXAMPLE II - Configuring a Single Local MTA

This example shows how to configure a single local MTA so that PRIME_X.400 becomes operational on a single node.

Stages in Configuring a Single Local MTA

1. Invoke CONFIG_X400 Command (see the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 1).
2. Define a Local MTA.
3. Configure Local Users.
4. Verify Configuration (see the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 5).
5. Save Configuration (see the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 6).
6. List Configuration (see the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 7).
7. Start Prime X.400 (see the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 8).

Figure 4-12 illustrates the hierarchy of screen forms that you navigate to Configure a Local MTA.

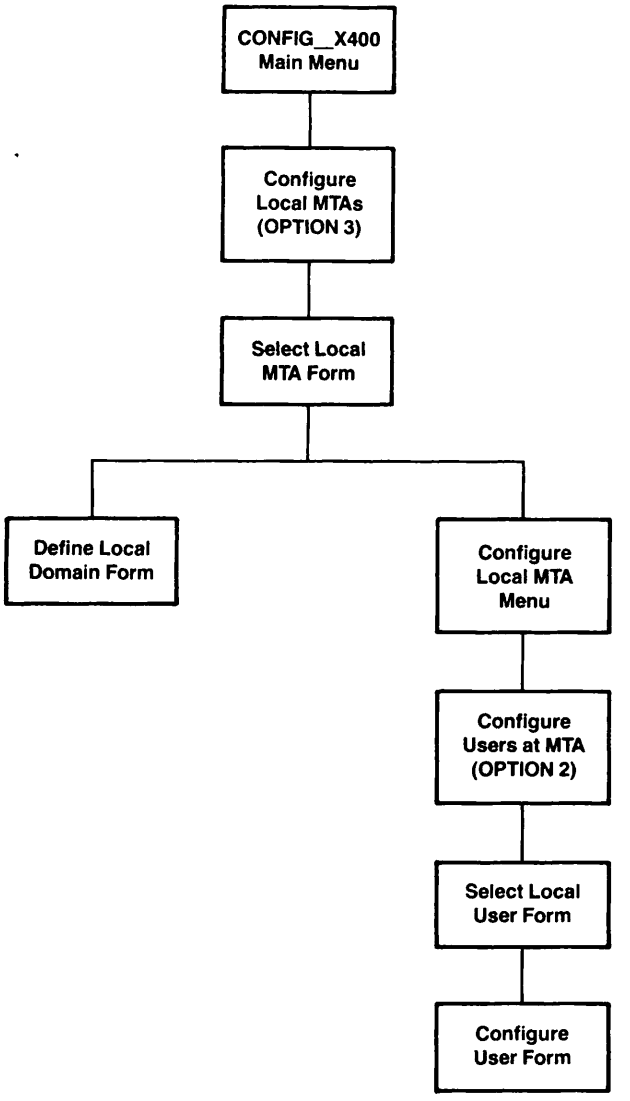

FIGURE 4-12. Hierarchy to Configure a Local MTA

Local MTAs

The Prime X.400 administrator is responsible for configuring Prime X.400 on a single node, or a group of local nodes, such as a local network. The MTAs, that the administrator defines within this group, are known as the local MTA group.

MTAs in the same local group share a set of default attributes, such as the type of logical link (association) between them, the default address space (domain), and inter-MTA passwords.

The following information can be optionally specified when configuring Local MTAs:

- Network addresses
- Links (associations) between them
- Passwords
- Service protocols and routing information
- MTA names
- Users' X.400 addresses and MTA attachments

Stage 1 - Invoke CONFIG_X400 Command

See the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 1.

Stage 2 - Define a Local MTA

Select option 3 - Configure Local MTAs from the CONFIG_X400 Main Menu and press TRANSMIT. You are presented with the Select Local MTA form as illustrated in Figure 4-13 (the `Select MTA:` field is blank until an MTA has been defined).

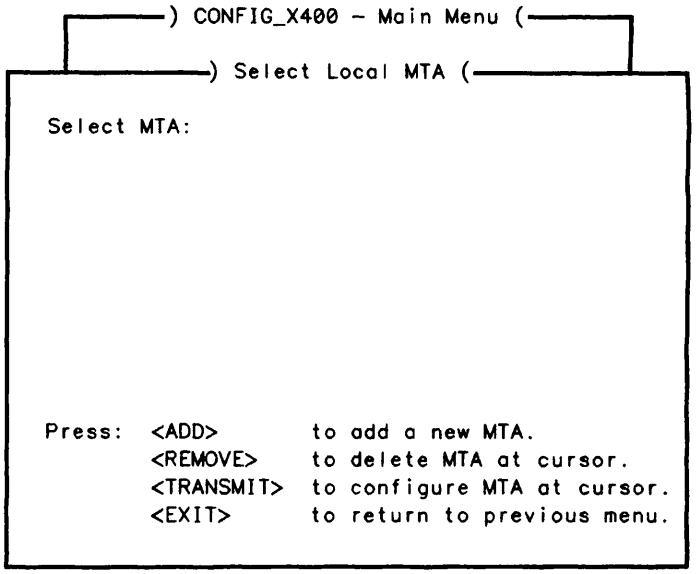

FIGURE 4-13. Select Local MTA Form

To define a Local MTA, press ADD. You are presented with the Define Local Domain Form, as illustrated in Figure 4-14.

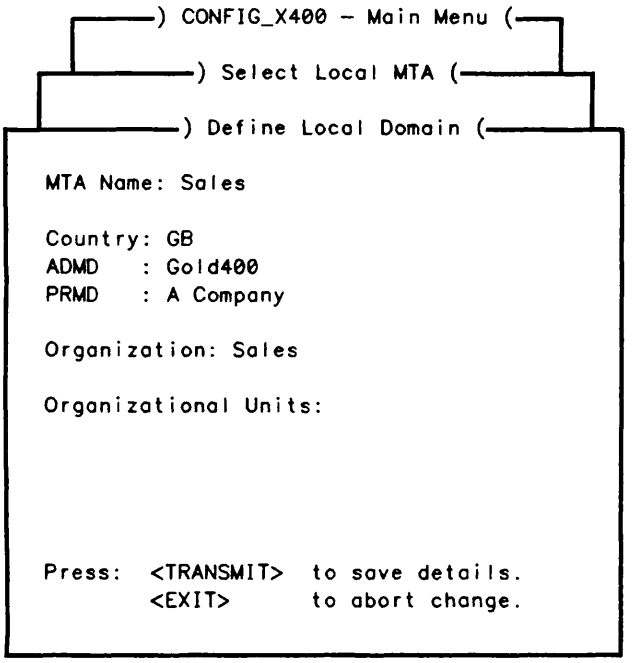

FIGURE 4-14. Define Local Domain Form

You must define the Local Domain by entering an MTA name and optionally, the Organization and Organizational Units.

Field	*Description*
MTA Name	Name of the Local MTA you wish to create.
Organization	A name assigned to an organization within a PRMD. It can contain a maximum of 64 characters.
Organizational Units	Names of units within an organization. Organizational units can contain a maximum of 32 characters.

Note

The fields Country, ADMD, and PRMD, on the define Local Domain Form, take the default values from the Configuration Domain Name and therefore cannot be altered.

When you have defined the local domain, the Select Local MTA form is resumed and displays the MTA that you have just created.

Stage 3 - Configure Local Users

Press the TRANSMIT key to select the MTA you have just created and the Configure Local MTA Menu is displayed as in Figure 4-15.

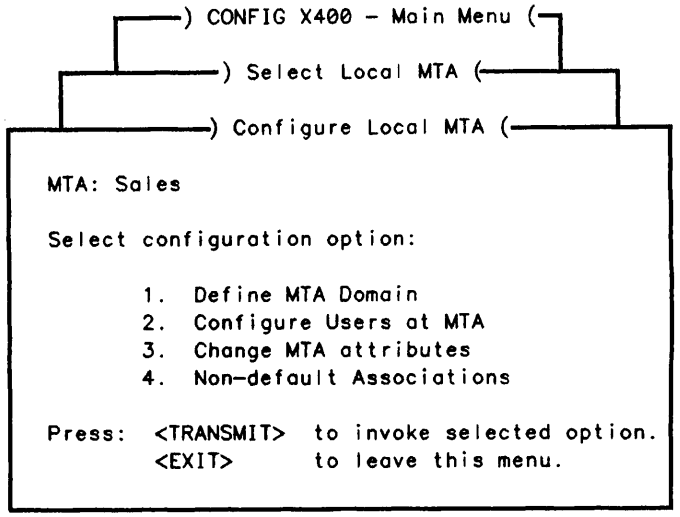

FIGURE 4-15. Configure Local MTA Menu

Select option 2 - Configure Users at MTA from the Configure Local MTA Menu, then press TRANSMIT. You are presented with the Select Local User Form as illustrated in Figure 4-16.

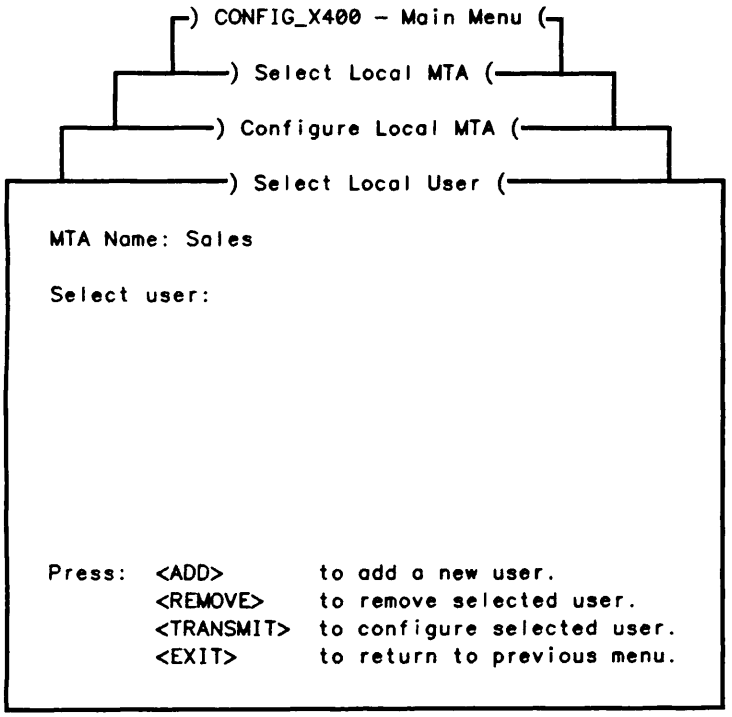

FIGURE 4-16. Select Local User Form

Initially this form is blank. Users are displayed in the `Select User:` field as you add them to the configuration.

To add a user, press the ADD key. You are presented with the Configure User Form, as illustrated in Figure 4-17.

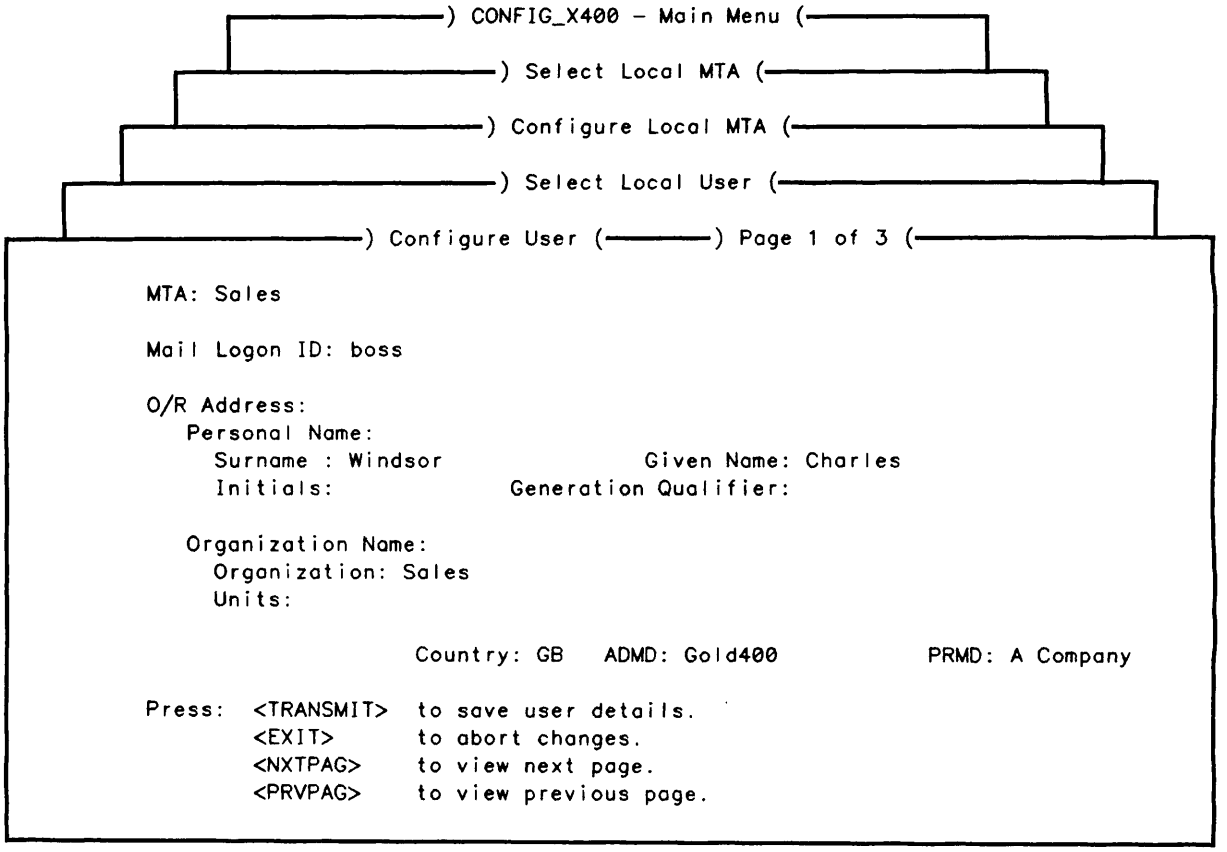

FIGURE 4-17. Configure User Form

O/R address components and their meanings are described in the following list. Country, ADMD, PRMD, and Organization Name, default to the values defined in the MTA's local domain name, if any.

Component	*Description*
Country	The country in which the network operates. This is set by the X.400 international control bodies. Specify either: • A 3-digit code as defined by CCITT X.121 (for example, the code for the UK is 234, for the US is 311) • A 2-letter code defined by ISO 3166/ALPHA-2 (for example, GB, US).
ADMD	Main administration domains within a country. If you are connecting to an X.400 service provided by a public carrier, use the name they give you. If you are connecting to a private mail application, either choose a name by bilateral agreement, or leave the field blank.

The ADMD name can contain a maximum of 16 printable characters.

PRMD — Private administration domains within a country. For mail applications that connect to ADMDs, the PRMD is a matter of negotiation with the ADMD administrator.

The PRMD name can contain a maximum of 16 printable characters.

Organization Name — A name assigned to an organization within a PRMD. It can contain a maximum of 64 printable characters.

Organizational Unit(s) — Names of units within an Organization. Up to 4 Organizational Units can be defined. Organizational Units can contain a maximum of 32 printable characters.

Personal Name — A collection of names that identify an individual mail service user:

Surname	:	40 characters.
Given Name	:	16 characters.
Initials	:	5 characters.
Generation Qualifier	:	3 characters.

When you have completed the form, press TRANSMIT (the select Local User Form is resumed).

The Select Local User Form displays the User you have just created. An example of the Select Local User Form is illustrated in Figure 4-18.

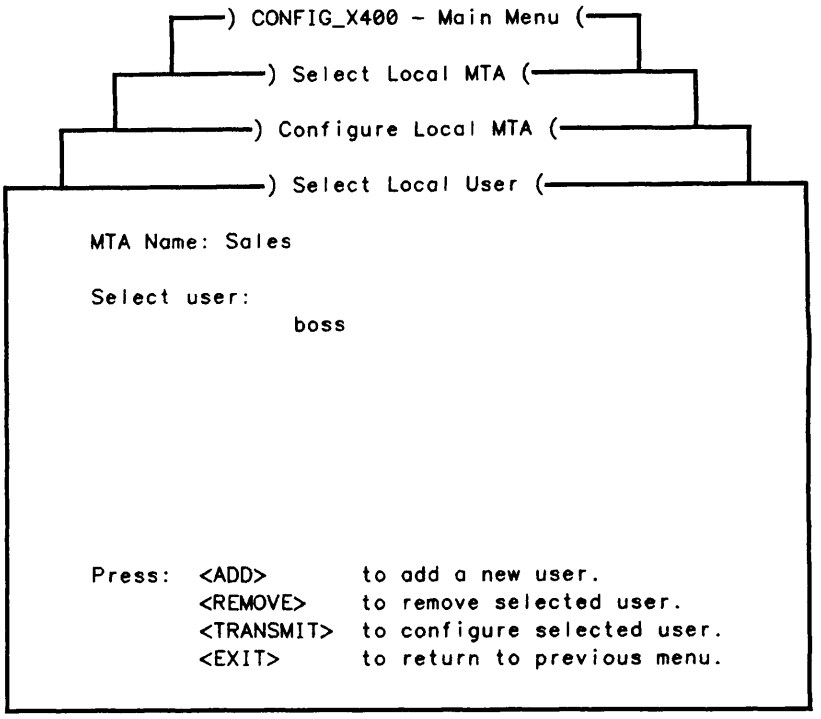

FIGURE 4-18. Select Local User Form

For each user you wish to enter, press ADD, and complete the Configure User Form.

Stage 4 - Verify Configuration

See the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 5.

Stage 5 - Save Configuration

See the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 6.

Stage 6 - List Configuration

See the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 7.

Stage 7 - Start Prime X.400

See the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 8.

EXAMPLE III - Adding a Local MTA

This example shows you how to add other Local MTAs.

Stages in Adding a Local MTA

1. Invoke CONFIG_X400 Command (see the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 1).
2. Define a Local MTA (see the previous section, EXAMPLE II - Configuring a Single Local MTA, Stage 2).
3. Configure Local Users (see the previous section, EXAMPLE II - Configuring a Single Local MTA, Stage 3).
4. Change the default association between adjacent MTAs, if necessary.
5. Verify Configuration (see the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 5).
6. Save Configuration (see the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 6).
7. List Configuration (see the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 7).
8. Start Prime X.400 (see the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 8).

Figure 4-19 illustrates the hierarchy of screen forms that you navigate to Add a Local MTA.

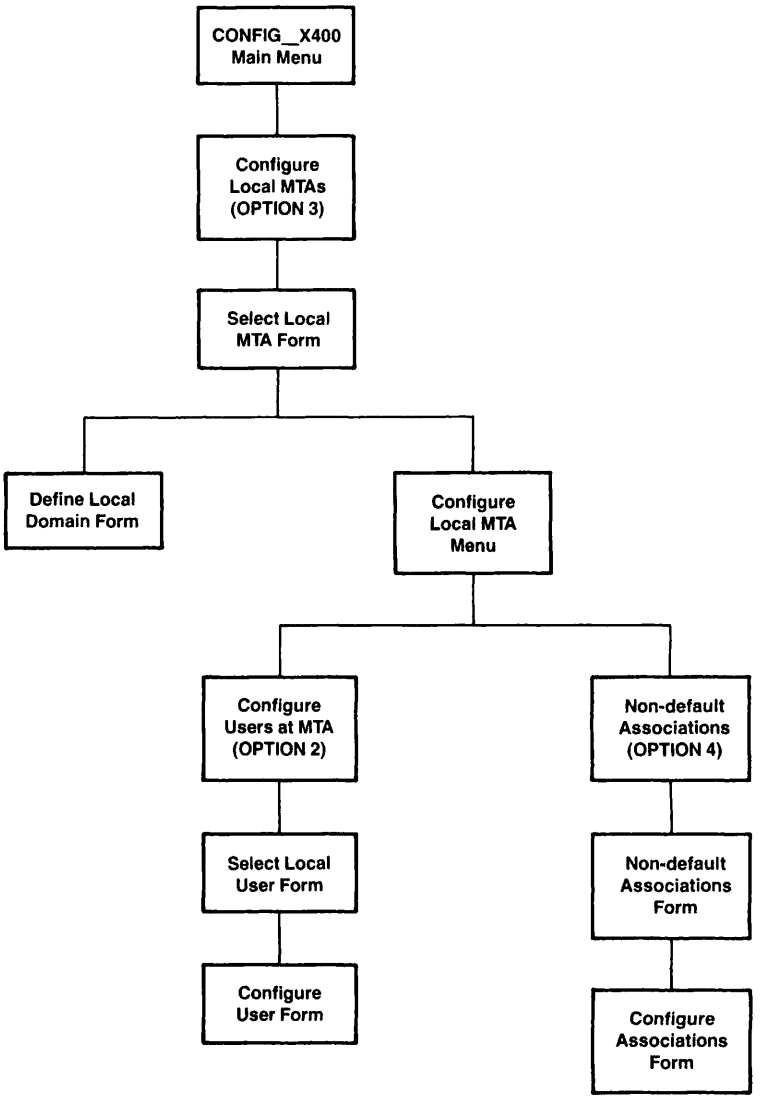

FIGURE 4-19. Hierarchy to Add a Local MTA

Stage 1 - Invoke CONFIG_X400 Command

See the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 1.

Stage 2 - Define a Local MTA

See the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 2.

Stage 3 - Configure Local Users

See the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 3.

Stage 4 - Changing Default Associations between MTAs

Select option 4, Non-default Associations from the Configure Local MTA Menu (Figure 4-15). You are presented with the Non-default Associations Form as illustrated in Figure 4-20.

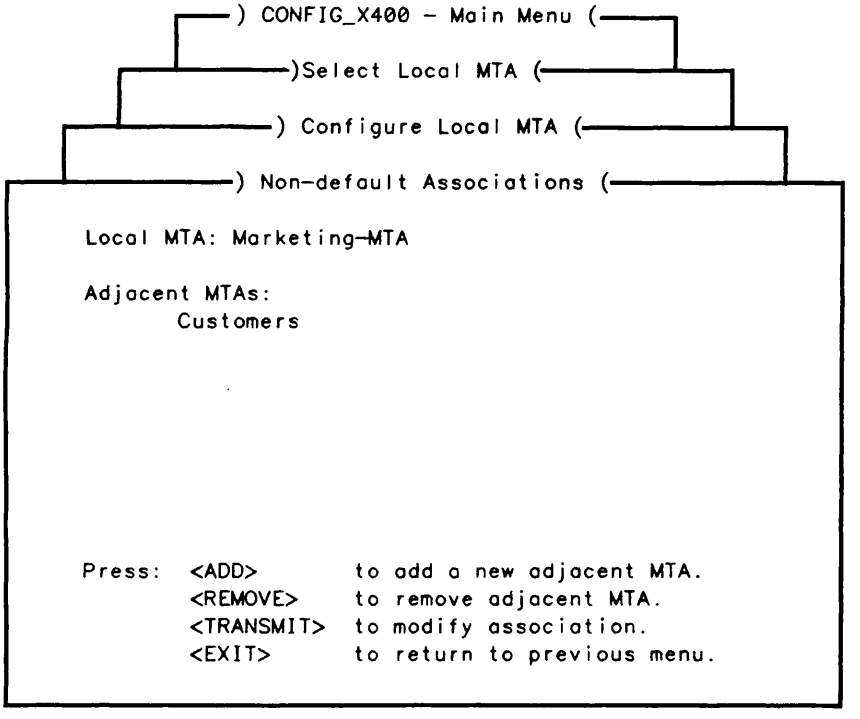

FIGURE 4-20. Non-default Associations Form

Press TRANSMIT to modify the association. You are presented with the Configure Associations Form as illustrated in Figure 4-21.

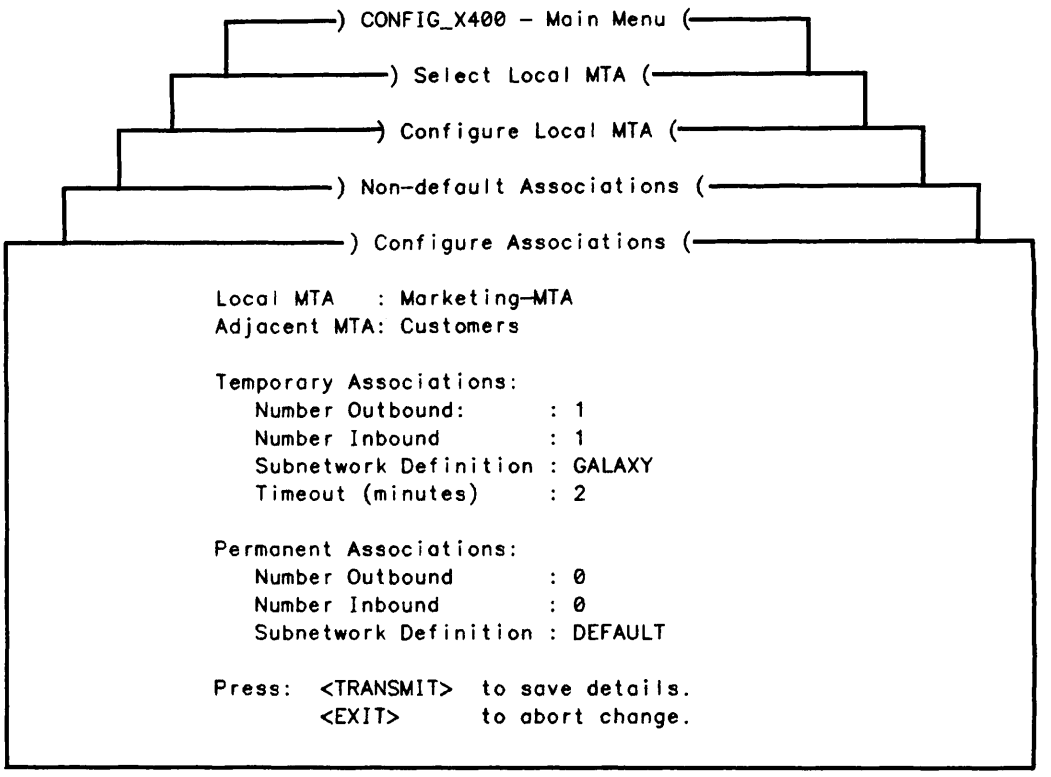

FIGURE 4-21. Configure Associations Form

Enter the subnetwork name (defined at Stage 3 of EXAMPLE I - Setting Configuration Defaults) in the Subnetwork Definition: field on this form and press TRANSMIT.

Stage 5 - Verify Configuration

See the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 5.

Stage 6 - Save Configuration

See the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 6.

Stage 7 - List Configuration

See the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 7.

Stage 8 - Start Prime X.400

See the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 8.

EXAMPLE IV - Configuring a Remote MTA

This example shows you how to configure a remote MTA.

Stages in Configuring a Remote MTA

1. Invoke CONFIG_X400 Command (see the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 1).
2. Define Remote MTA.
3. Define Remote MTA O/R Address Space.
4. Define Remote MTA Attributes.
5. Define an association between a Local MTA and the Remote MTA.
6. Verify Configuration (see the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 5).
7. Save Configuration (see the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 6).
8. List Configuration (see the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 7).
9. Start Prime X.400 (see the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 8).

Figure 4-22 illustrates the hierarchy of screen forms that you navigate to Configure a Remote MTA.

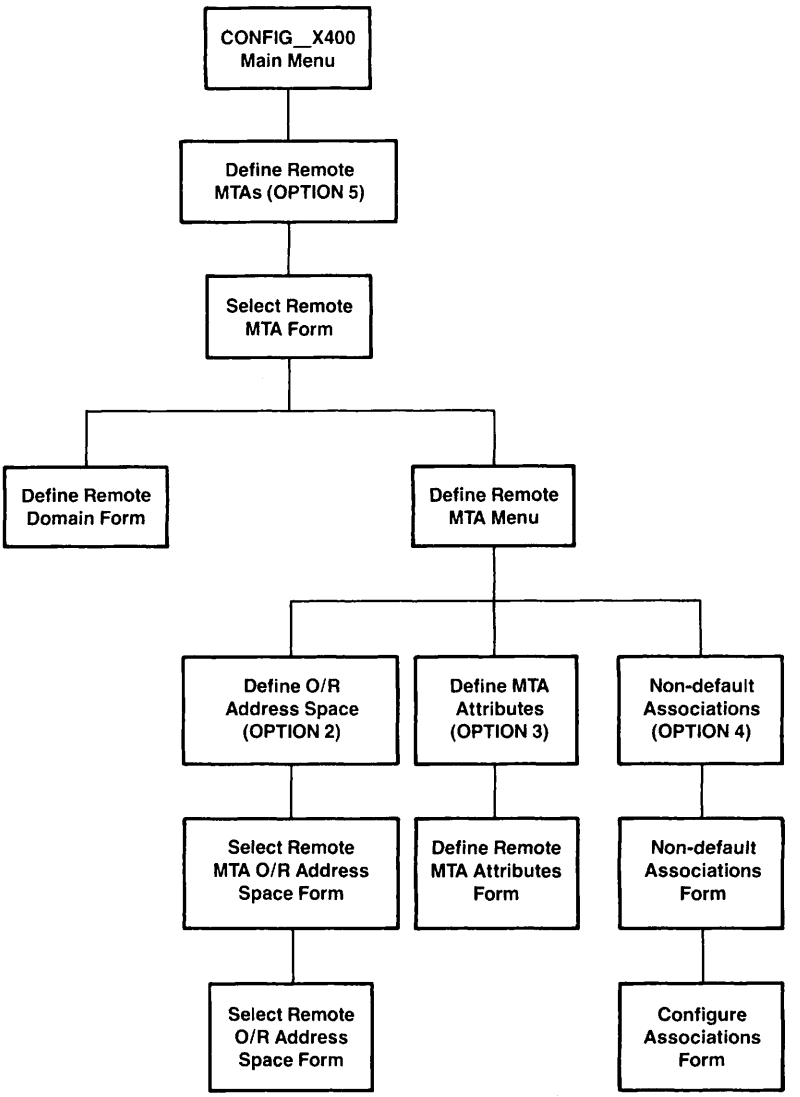

FIGURE 4-22. Hierarchy to Configure a Remote MTA

Remote MTAs

Remote MTAs are MTAs outside your immediate control. Remote MTAs act as gateways to other user domains on the X.400 network.

The following information can be optionally specified when defining Remote MTAs:

- Network addresses

- Protocols for communicating with the local MTA group
- Remote MTA's domain name
- Remote user address space (domain) associated with the MTA

Stage 1 - Invoke CONFIG_X400 Command

See the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 1.

Stage 2 - Define a Remote MTA

Select option 5, `Define Remote MTAs`, from the CONFIG_X400 Main Menu (Figure 4-4) and press TRANSMIT. You are presented with the Select Remote MTA form as illustrated in Figure 4-23. (The `Select MTA:` field is blank until a remote MTA has been defined).

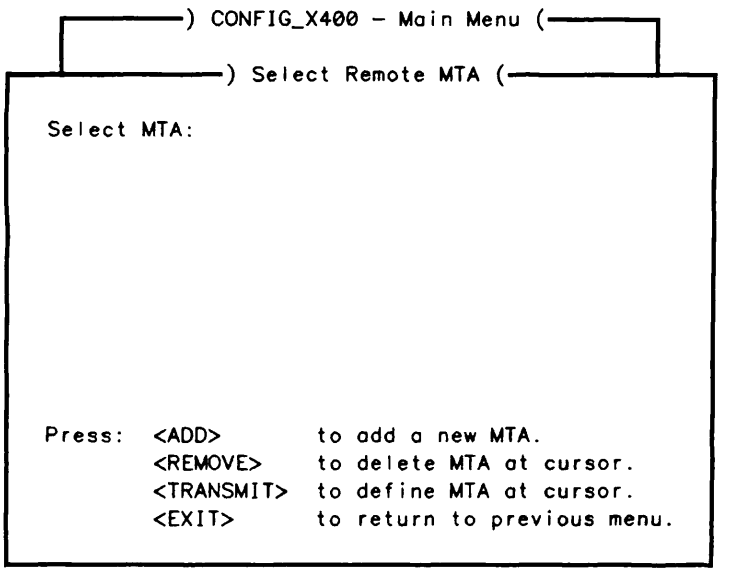

FIGURE 4-23. Select Remote MTA Form

To define a remote MTA, press ADD. You are presented with the Define Remote Domain Form as illustrated in Figure 4-24.

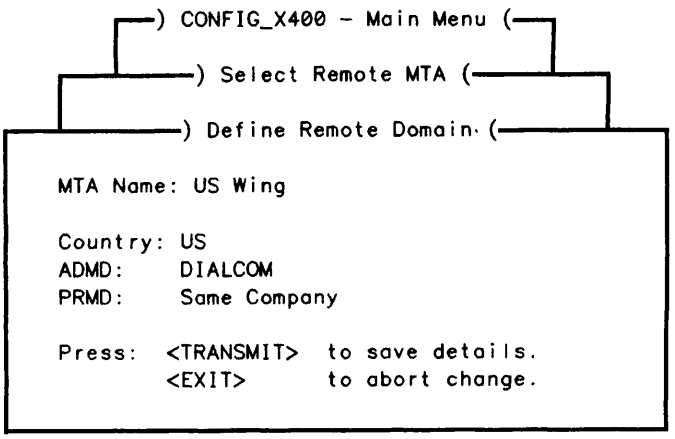

FIGURE 4-24. Define Remote Domain Form

You must define the Domain by entering an MTA Name, Country, ADMD, and PRMD (details of how to complete these fields can be found in the previous section, EXAMPLE II - Configuring a Single Local MTA, Stage 2).

When you have defined the domain, press TRANSMIT. The Select Remote MTA form is resumed and displays the MTA you have just created. Use the ADD key to enter more remote MTAs as required.

Stage 3 - Define Remote MTA O/R Address Space

Select an MTA from the Select Remote MTA Form and you are presented with the Define Remote MTA Menu as illustrated in Figure 4-25.

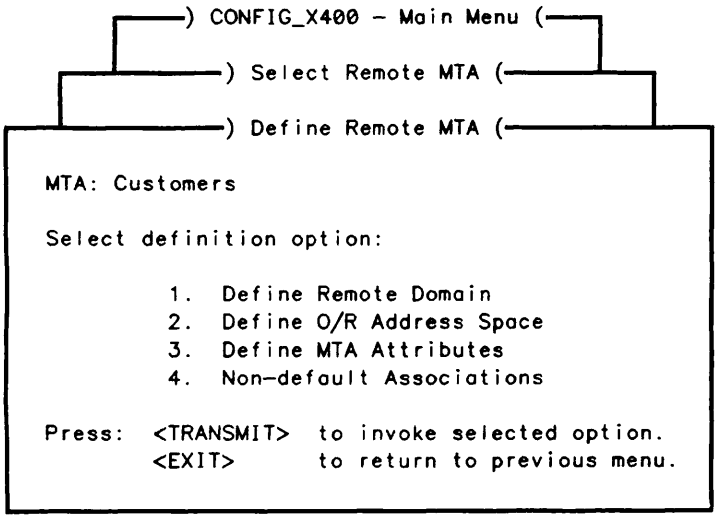

FIGURE 4-25. Define Remote MTA Menu

You must now define the O/R addresses accessible via the MTA and other MTA attributes. Select option 2, `Define O/R Address Space`. You are presented with the Select Remote MTA O/R Address Space form as illustrated in Figure 4-26.

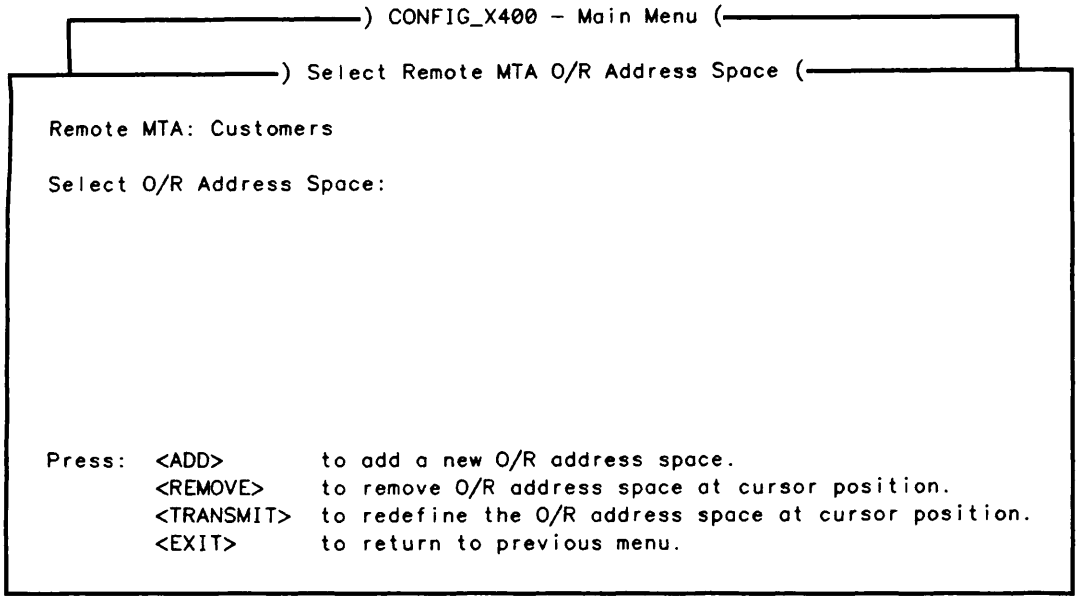

FIGURE 4-26. Select Remote MTA O/R Address Space Form

To enter an O/R address for the remote MTA, press ADD. You are presented with the Select Remote O/R Address Space form as illustrated in Figure 4-27.

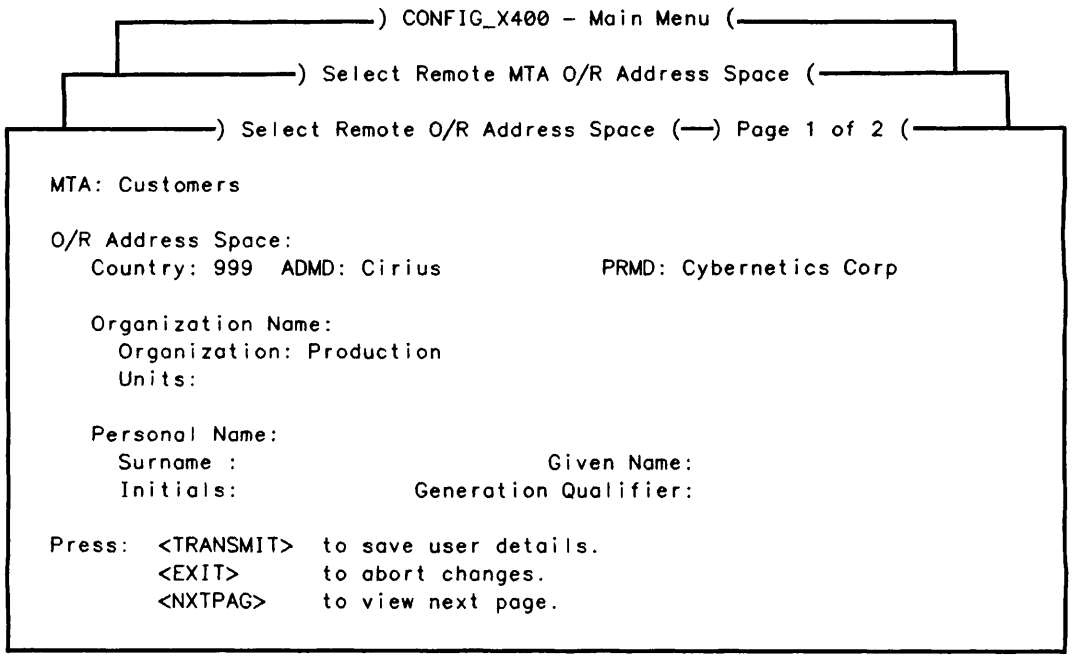

FIGURE 4-27. Select Remote O/R Address Space Form

It is not necessary to enter a full O/R address, simply supply sufficient attributes to uniquely identify the O/R addresses that can be reached via the MTA (O/R address components and their meanings can be found earlier in this Chapter). When you have completed the form, press TRANSMIT.

The Select Remote MTA O/R Address Space form is resumed. Use ADD to enter other O/R addresses.

Stage 4 - Define Remote MTA Attributes

When you have entered the O/R addresses for the MTA, select option 3, `Define MTA Attributes` from the Define Remote MTA Menu.

You are presented with the Define Remote MTA Attributes form as illustrated in Figure 4-28.

Remote MTA attributes are described in the following list:

Attribute	*Description*
MTA Name	The name you must use to communicate with the remote MTA. Obtain the name from the MTA's administrator.
MTA's Password	The password that identifies the remote MTA. Obtain the password from the remote MTAs' administrator.

Our Password

The password that the remote MTA must supply in order to communicate with the local MTA. It is assigned locally.

Protocol Type

The protocol type used by the remote MTA. Set to either MOTIS or CCITT-84. The default is CCITT-84.

Network Address

The ISO network address (NSAP) of the MTA. Specify the address in one of the following formats:

- Nodename
- 15 digits preceded by a colon (X121 address)

If in doubt, allow the default to be selected.

Transport Protocol Selector

Selects the ISO transport protocol. This should be obtained from the remote MTAs administrator.

Enter as 2 hexadecimal digits per byte. For example, if the application uses printable characters, and you agree on the 2-byte code that corresponds to ASCII 17, then you must enter 3137.

The maximum permitted size is 32 bytes, and the default value is 3432.

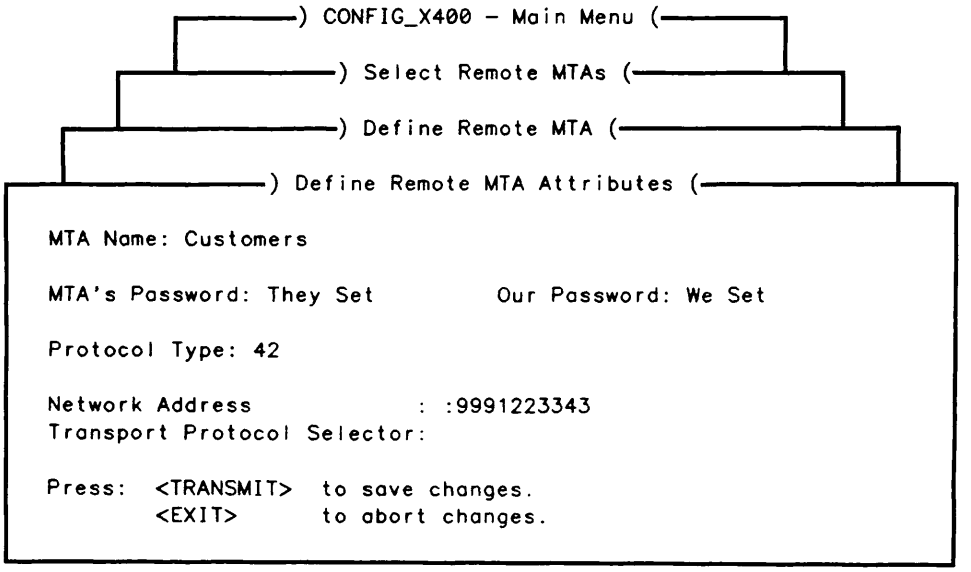

FIGURE 4-28. Define Remote MTA Attributes Form

When you have completed the form, press TRANSMIT.

Stage 5 - Define Association between a Local MTA and the Remote MTA

When all remote MTAs are defined, the associations between them and the local MTAs must be set up.

Select option 4, `Non-default Associations` from the Define Remote MTA Menu (Figure 4-25). You are presented with the Non-default Associations Form, as illustrated in Figure 4-29.

```
) CONFIG_X400 - Main Menu (
)Select Remote MTA (
) Define Remote MTA (
) Non-default Associations (

Local MTA: Marketing-MTA

Adjacent MTAs:




Press:  <ADD>       to add a new adjacent MTA.
        <REMOVE>    to remove adjacent MTA.
        <TRANSMIT>  to modify association.
        <EXIT>      to return to previous menu.
```

FIGURE 4-29. Define Non-default Associations Form

Press the ADD key. You are presented with a list of all possible MTAs to which the selected MTA may be associated.

Position the cursor to the desired Local MTA and press TRANSMIT. Pressing TRANSMIT saves the Local MTA and resumes the Define Non-default Associations form.

Further local MTAs can be added by pressing the ADD key.

Stage 6 - Verify Configuration

See the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 5.

Stage 7 - Save Configuration

See the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 6.

Stage 8 - List Configuration

See the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 7.

Stage 9 - Start Prime X.400

See the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 8.

EXAMPLE V - Configuring Large Numbers of Local Users

This example shows how to Configure Large Numbers of Local Users.

Stages in Configuring Large Numbers of Local Users

1. Use CONFIG_X400 Command to create a configuration with all required local MTAs defined (see the previous sections, EXAMPLE I - Setting Configuration Defaults, EXAMPLE II - Configuring a Single Local MTA, and EXAMPLE III - Adding a Local MTA).
2. Edit the text configuration file to add local users.
3. Invoke CONFIG_X400 Command using configuration just created (see the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 1).
4. Verify Configuration (see the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 5).
5. Save Configuration (see the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 6).
6. List Configuration (see the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 7).
7. Start Prime X.400 (see the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 8).

Figure 4-30 illustrates the hierarchy of screen forms that you navigate to enter large numbers of users.

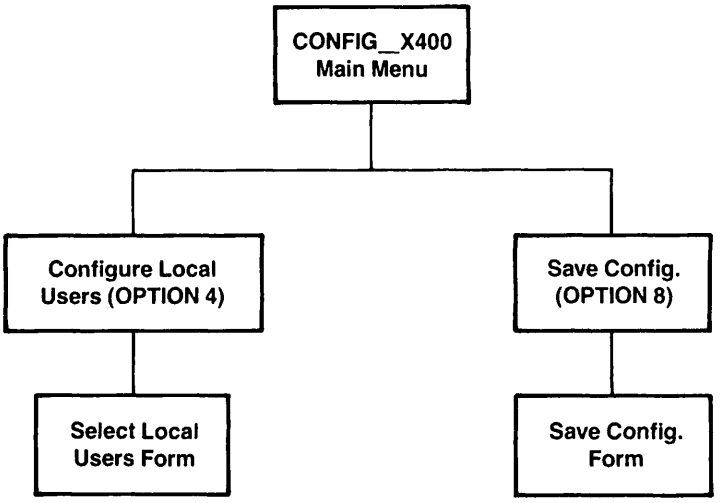

FIGURE 4-30. Hierarchy to Configure Large Numbers of Users

Stage 1 - Invoke CONFIG_X400 Command

See the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 1.

Stage 2 - Editing the Text Configuration File

The .CONFIG file is a text file. An example .CONFIG file is illustrated in Figure 4-31.

```
#version 2
#date 586092229
#user X400_ADMIN
#node ADMIN
#comment Config ready for bulk load of users
#nln 6
#file EXAMPLE.CONFIG
L "A Company".Gold400.GB
M Development -RTS 0 3 1024 1024 -PR 8823 -AD X121."Development-MTA".3432. -DO "A Company"
M Marketing -RTS 0 3 1024 1024 -PR 8823 -AD X121."Marketing-MTA".3432. -DO %Marketing"A Co
M Sales -RTS 0 3 1024 1024 -PR 8823 -AD X121."Sales-MTA".3432. -DO %Sales"A Company".Gold4
AS $LOCAL 0 1 15 DEFAULT DEFAULT
AS $REMOTE 0 0 2 DEFAULT DEFAULT
U boss@Sales Charles.Windsor%Sales"A Company".Gold400.GB
U servant@Sales Alfred.Hudson%Sales"A Company".Gold400.GB
AR boss
```

FIGURE 4-31. .CONFIG File Example

If you have an online list of users that need to be entered into the configurator (such as a telephone list), then the list can be modified to a form suitable for the configurator, using your favourite editor.

Create a list, one user per line, using the following format as a guide:

```
U <Mail_id>@<MTA>  <or address>
```

and insert it at the bottom of the .CONFIG file. Figure 4-32 illustrates the .CONFIG file with a list of users attached.

```
#version 2
#date 586092229
#user X400_ADMIN
#nodeADMIN
#comment Config ready for bulk load of users
#nln 6
#file EXAMPLE.CONFIG
L "A Company".Gold400.GB
M Development -RTS 0 3 1024 1024 -PR 8823 -AD X121."Development-MTA".3432. -DO "A Company"
M Marketing -RTS 0 3 1024 1024 -PR 8823 -AD X121."Marketing-MTA".3432. -DO %Marketing"A Co
M Sales -RTS 0 3 1024 1024 -PR 8823 -AD X121."Sales-MTA".3432. -DO %Sales"A Company".Gold4
AS $LOCAL 0 1 15 DEFAULT DEFAULT
AS $REMOTE 0 0 2 DEFAULT DEFAULT
U boss@Sales Charles.Windsor%Sales"A Company".Gold400.GB
U servant@Sales Alfred.Hudson%Sales"A Company".Gold400.GB
AR bossU master@Development Christopher.Robin%
U thinker@Development Pooh.Bear%
U seller@Marketing Marvin.Android%
```

FIGURE 4-32. CONFIG file with Users Attached Example

Note

Only the personal name attributes of the new users have been entered in the Figure 4-32 example. The full O/R address is constructed by the configurator from the MTA domain name.

It is important that you enter the % character at the end of each personal name.

Stage 3 - Invoking CONFIG_X400 Command to Check Users

Invoke the CONFIG_X400 command using the .CONFIG file just created in Stage 2, and select option 4, `Configure Local Users` from the CONFIG_X400 Main Menu. You are presented with the Select Local Users form as illustrated in Figure 4-33. Use the screen to check that you have loaded the users that you think you have.

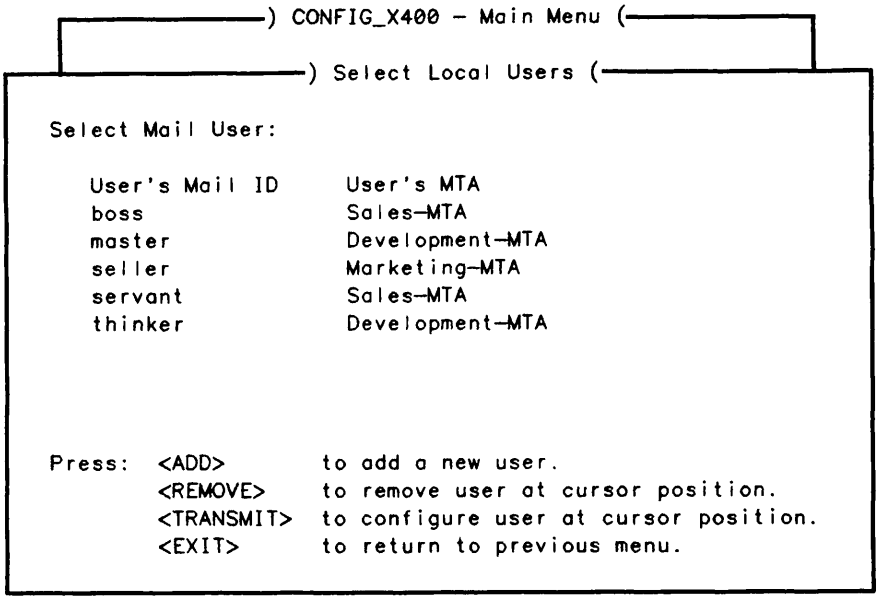

FIGURE 4-33. Select Local Users Form

Stage 4 - Verify Configuration

See the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 5.

Stage 5 - Save Configuration

See the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 6.

Stage 6 - List Configuration

See the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 7.

Stage 7 - Start Prime X.400

See the previous section, EXAMPLE I - Setting Configuration Defaults, Stage 8.

APPENDICES

A

INSTALLATION AND DIAGNOSTICS

Installation

To install Prime X.400 on the system,

- Mount the tape.
- Restore the tape using MAGRST.
- Run the program PRIME_X400>PRIME_X400.INSTALL.CPL.

The install program builds the PRIME_X400* runtime directory, updates the system's CMDNC0, and LIBRARIES*, HELP*, and LIB directories, and updates the system search rules.

If a Rev.2 (or later) Priforma is not already installed on your system, then this must also be installed from the supplied tape.

The PRIME_X400* Directory

PRIME_X400* is the runtime product directory. It holds the server runfiles security files and miscellaneous data files, and contains subdirectories for the routing tables, journal log files, and other essential data files.

Files and subdirectories in PRIME_X400* are described as follows:

File	*Contents*
X400_[DATE].COMO	The como file for the Prime X.400 server. This contains startup information, subsystem error messages, and diagnostic information.
X400_SERVER.RUN	The program that runs the Prime X.400 server.
X400_SERVER.CPL	The CPL file that controls server initialization.

X400_CACHE.RUN	The program that builds the runtime routing tables.
X400_SECURITY.RUN	The program that sets the ACL's for Prime X.400 logon IDs.
PRIME_X400.CONFIG	The Prime X.400 default configuration file.
X409.TXT	Contains parameters used in X.409 message parsing.
CST.INP	Contains parameters used by the ADMIN_X400 command interface.
SCT.INP	Contains parameters used by the Prime X.400 server.
ADMIN.ACAT	The access category that controls access to the ADMIN_X400 command.

Subdirectory	*Contents*
OBJECTS	Contains files OMRDTABL, which is the runtime routing table and the data files for all messages being processed.
EVENQ	This directory contains backup information used to recover mail files after system crashes etc.
ODDQ	This directory contains backup information used to recover mail files after system crashes etc.
SECURITY	Defines ACLs for Prime X.400 logon IDs.
ADMIN	Contains the help file for the ADMIN_X400 command.
X400_CACHE	Contains local language help text and messages.
FORMS	Contains the data files for the CONFIG_X400 forms.

Error and Event Logging

Error Logging

Errors from the Prime X.400 subsystem software, and major events such as startup, are logged in the server COMO file.

Error messages are recorded in a standard format. For example:

```
08 Jun 88 17:43:06  Product ID  0001     Severity 000C  Error Code F006
                    User Data 1 0013  User Data 2 0000
```

Product ID is an internal identifier that indicates which software module generated the error.

Severity is an indication of the importance of the event to Prime X.400 operation. This is a value 1 (least severe) through hexadecimal F (most severe).

Error Code is the error reference number.

User Data (1 and 2) are parameters displayed in the message.

For explanations of the error codes, and meanings of product and severity codes, see Appendix B.

For details of User Data codes, see your Prime representative.

B

PRIME X.400 ERROR MESSAGES

Error Messages

This appendix lists and explains the error messages produced by Prime X.400.

Error: 0001
Mismatch in CST Version numbers. This is caused by the old version of the CST being used by an updated ADMIN_X400. The CST compiler should be rebuilt and rerun. This error is fatal. User Data 1 contains the ADMIN_X400 Version Number. User Data 2 contains the CST Compiler Version Number (ADMIN_X400).

Error: 0002
There is no entry for ADMIN_X400 in the CSUT table. This error is fatal. (ADMIN_X400).

Error: 0003
The console type is incompatible with the CSUT entries, e.g., a SNAP-5 Operator Console with no SNAP-5 CSUT entry. This error is fatal. (ADMIN_X400).

Error: 0100
A Data-CSI (CS-Msg) was sent by a Service User specifying an unrecognised message source. This error is low severity. User Data 1 contains the Message Source. User Data 2 contains the Message Number (ADMIN_X400).

Error: 0101
A Data-CSI (CS-Msg) was sent with an unknown message number. This error is low severity. User Data 1 contains the Message Source. User Data 2 contains the Message Number (ADMIN_X400).

Error: 0102
A Data-CSI (CS-Msg) was sent to a console which was detached or not configured for the sending Service User. This error is low severity. (ADMIN_X400).

Error: 0300
No Console Usage Records specified in SCT; there must be at least one. This error is fatal. (ADMIN_X400).

Error: 0301
The maximum number of consoles allowed per Service User has been exceeded by the SCT. This error is fatal. (ADMIN_X400).

Error: 0302
The console type specified in an SCT Console Usage Record is invalid. This error is fatal. (ADMIN_X400).

Error: 0303
The automatic attach flag specified in an SCT Console Usage Record is invalid. This error is fatal. (ADMIN_X400).

Error: 0304
The broadcast flag specified in an SCT Console Usage Record is invalid. This error is fatal. (ADMIN_X400).

Error: 0305
The maximum number of consoles allowed has been exceeded by the SCT. This error is fatal. (ADMIN_X400).

Error: 0306
The SCT has specified that a Service User has multiple usages of the same console - this is not permitted. This error is fatal. (ADMIN_X400).

Error: 0307
The maximum number of usages allowed per console has been exceeded. This error is fatal. (ADMIN_X400).

Error: 0500
ADMIN_X400 cannot initialize. SAM has reported that the product is not configured in the SCT. This error is fatal. (ADMIN_X400).

Error: 0601
ADMIN_X400 has received an invalid message on the CSI while initializing. This error is low severity. (ADMIN_X400).

Error: 0602
ADMIN_X400 has received an invalid message on the CSI during normal operating. This error is low severity. (ADMIN_X400).

Error: 0603
ADMIN_X400 has received a message for a closed connection. This error is low severity. (ADMIN_X400).

Error: 0604
ADMIN_X400 has received a message with a bad source LPI. This error is low severity. (ADMIN_X400).

Error: 0605
ADMIN_X400 has received a message with a destination I which is out of range. This error is low severity. (ADMIN_X400).

Error: 0606
ADMIN_X400 has received a message while waiting for a Close Response. This error is low severity. (ADMIN_X400).

Error: 0607
ADMIN_X400 has received a message with a bad Routing Code. This error is low severity. (ADMIN_X400).

Error: 0900
The PMOD returned an illegal console file number. This error is fatal. User Data 1 contains the Console File Number (ADMIN_X400).

Error: 0901
The PMOD returned an illegal result type. This error is fatal. User Data 1 contains the Console File Number (ADMIN_X400).

Error: 0902
The PMOD returned a result for a closed read console file. This error is fatal. User Data 1 contains the Console File Number (ADMIN_X400).

Error: 0903
The PMOD returned a result for a closed write console file. This error is fatal. User Data 1 contains the Console File Number (ADMIN_X400).

Error: 0904
The PMOD returned a NIL pointer in an SCPPREAD result. This error is fatal. User Data 1 contains the Console File Number (ADMIN_X400).

Error: 0905
The PMOD returned a NIL pointer in an SCPPWRIT or SCPPOPEN result for a write console file. This error is fatal. User Data 1 contains the Console File Number (ADMIN_X400).

Error: 0906
The PMOD returned an SCPPREAD result, with OFFSET less than 1. This error is fatal. User Data 1 contains the Console File Number (ADMIN_X400).

Error: 0907
The PMOD returned an SCPPWRIT or SCPPOPEN result on a write console file, with OFFSET less than 1. This error is fatal. User Data 1 contains the Console File Number (ADMIN_X400).

Error: 0908
The PMOD returned an SCPPREAD result, with LENGTH less than 0. This error is fatal. User Data 1 contains the Console File Number (ADMIN_X400).

Error: 0909
The PMOD returned an SCPPWRIT or SCPPOPEN result on a write console file, with LENGTH less than 0. This error is fatal. User Data 1 contains the Console File Number (ADMIN_X400).

Error: 0910
The PMOD returned an SCPPREAD result with OFFSET + LENGTH - 1 > SCDATALN. This error is fatal. User Data 1 contains the Console File Number (ADMIN_X400).

Error: 0911
The PMOD returned an SCPPWRIT or SCPPOPEN result on a write console file with OFFSET + LENGTH - 1 > SCDATALN. This error is fatal. User Data 1 contains the Console File Number (ADMIN_X400).

Error: 07C1
Mismatch in CST Version numbers. This is caused by the old version of the CST being used by an updated ADMIN_X400. The CST compiler should be rebuilt and rerun. This error is fatal. User Data 1 contains the ADMIN_X400 Version Number. User Data 2 contains the CST Compiler Version Number (ADMIN_X400).

Error: 07C2
There is no entry for ADMIN_X400 in the CSUT table. This error is fatal. (ADMIN_X400).

Error: 07C3
The console type is incompatible with the CSUT entries, e.g., a SNAP-5 Operator Console with no SNAP-5 CSUT entry. This error is fatal. (ADMIN_X400).

Error: 0741
A Data-CSI (CS-Msg) was sent by a Service User specifying an unrecognised message source. This error is low severity. User Data 1 contains the Message Source. User Data 2 contains the Message Number (ADMIN_X400).

Error: 0742
A Data-CSI (CS-Msg) was sent with an unknown message number. This error is low severity. User Data 1 contains the Message Source. User Data 2 contains the Message Number (ADMIN_X400).

Error: 0743
A Data-CSI (CS-Msg) was sent to a console that was detached, or not configured for the sending Service User. This error is low severity. (ADMIN_X400).

Error: 07C4
No Console Usage Records specified in SCT; there must be at least one. This error is fatal. (ADMIN_X400).

Error: 07C5
The maximum number of consoles allowed per Service User has been exceeded by the SCT. This error is fatal. (ADMIN_X400).

Error: 07C6
The console type specified in an SCT Console Usage Record is invalid. This error is fatal. (ADMIN_X400).

Error: 07C7
The automatic attach flag specified in an SCT Console Usage Record is invalid. This error is fatal. (ADMIN_X400).

Error: 07C8
The broadcast flag specified in an SCT Console Usage Record is invalid. This error is fatal. (ADMIN_X400).

Error: 07C9
The maximum number of consoles allowed has been exceeded by the SCT. This error is fatal. (ADMIN_X400).

Error: 07CA
The SCT has specified that a Service User has multiple usages of the same console; this is not permitted. This error is fatal. (ADMIN_X400).

Error: 07CB
The maximum number of usages allowed per console has been exceeded. This error is fatal. (ADMIN_X400).

Error: 07CC
ADMIN_X400 cannot initialize. SAM has reported that the product is not configured in the SCT. This error is fatal. (ADMIN_X400).

Error: 0744
ADMIN_X400 has received an invalid message on the CSI while initializing. This error is low severity. (ADMIN_X400).

Error: 0745
ADMIN_X400 has received an invalid message on the CSI during normal operating. This error is low severity. (ADMIN_X400).

Error: 0746
ADMIN_X400 has received a message for a closed connection. This error is low severity. (ADMIN_X400).

Error: 0747
ADMIN_X400 has received a message with a bad source LPI. This error is low severity. (ADMIN_X400).

Error: 0748
ADMIN_X400 has received a message with a destination I, which is out of range. This error is low severity. (ADMIN_X400).

Error: 0749
ADMIN_X400 has received a message while waiting for a Close Response. This error is low severity. (ADMIN_X400).

Error: 074A
ADMIN_X400 has received a message with a bad Routing Code. This error is low severity. (ADMIN_X400).

Error: 07CD
The PMOD returned an illegal console file number. This error is fatal. User Data 1 contains the Console File Number (ADMIN_X400).

Error: 07CE
The PMOD returned an illegal result type. This error is fatal. User Data 1 contains the Console File Number (ADMIN_X400).

Error: 07CF
The PMOD returned a result for a closed read console file. This error is fatal. User Data 1 contains the Console File Number (ADMIN_X400).

Error: 07D0
The PMOD returned a result for a closed write console file. This error is fatal. User Data 1 contains the Console File Number (ADMIN_X400).

Error: 07D1
The PMOD returned a NIL pointer in an SCPPREAD result. This error is fatal. User Data 1 contains the Console File Number (ADMIN_X400).

Error: 07D2
The PMOD returned a NIL pointer in an SCPPWRIT or SCPPOPEN result for a write console file. This error is fatal. User Data 1 contains the Console File Number (ADMIN_X400).

Error: 07D3
The PMOD returned an SCPPREAD result with OFFSET less than 1. This error is fatal. User Data 1 contains the Console File Number (ADMIN_X400).

Error: 07D4
The PMOD returned an SCPPWRIT or SCPPOPEN result on a write console file with OFFSET less than 1. This error is fatal. User Data 1 contains the Console File Number (ADMIN_X400).

Error: 07D5
The PMOD returned an SCPPREAD result with LENGTH less than 0. This error is fatal. User Data 1 contains the Console File Number (ADMIN_X400).

Error: 07D6
The PMOD returned an SCPPWRIT, or SCPPOPEN, result on a write console file, with LENGTH less than 0. This error is fatal. User Data 1 contains the Console File Number (ADMIN_X400).

Error: 07D7
The PMOD returned an SCPPREAD result, with OFFSET + LENGTH - 1 > SCDATALN. This error is fatal. User Data 1 contains the Console File Number (ADMIN_X400).

Error: 07D8
The PMOD returned an SCPPWRIT, or SCPPOPEN, result on a write console file, with OFFSET + LENGTH - 1 > SCDATALN. This error is fatal. User Data 1 contains the Console File Number (ADMIN_X400).

Error: 0A00
Product not supported. This error is low severity. (SAM).

Error: 0A01
Product not found in SCT. This error is low severity. (SAM).

Error: 0A02
SNAP-3270 Port not found. This error is low severity. (SAM).

Error: 0A03
Application not found. This error is low severity. (SAM).

Error: 0A04
SNAP-RJE Workstation not found. This error is low severity. (SAM).

Error: 0A05
PLU not found. This error is low severity. User Data 1 contains the Source Product. (SAM).

Error: 0A06
Destination name not found in SCT. This error is low severity. (SAM).

Error: 0A07
SNAP-5 Console Group not found. This error is low severity. (SAM).

Error: 0A08
Invalid message discarded. This error is low severity. (SAM).

Error: 0A09
Resource ID incorrect. This error is low severity. (SAM).

Error: 0A40
Open () Request not recognized. This error is severe. User Data 1 contains the open type. (SAM).

Error: 0A41
No element on helper message. This error is severe. (SAM).

Error: 0A42
Product Locality not found for a helper message. This error is severe. (SAM).

Error: 0A43
Repeated RRN in SCT. This error is severe. User Data 1 contains the SCT index. (SAM).

Error: 0A44
SCT RRN too big. This error is severe. User Data 1 contains the SCT index. (SAM).

Error: 0A45
Locality not found by Open locality request handler. This error is severe. (SAM).

Error: 0A46
Application is not HLI Type. This error is severe. (SAM).

Error: 0B00
SAM cannot obtain enough elements. This error is low severity. User Data 1 contains the bytes required. User Data 2 contains the elements left. (X.400 Server).

Error: 0B01
Base Helper already open. This error is low severity. (X.400 Server).

Error: 0B02
An attempt to new a variable has failed. Var type specifies: 1-Buffer header, 2-Buffer element, 3-Correlation Table entry. This error is low severity. User Data 1 contains the var type. (X.400 Server).

Error: 0B40
Message routed to dummy AMOD. This error is severe. User Data 1 contains the Amod Index. (X.400 Server).

Error: 0B41
Path error during Initialization. This error is severe. (X.400 Server).

Error: 0B42
Path error after Initialization. This error is severe. User Data 1 contains the error type. User Data 2 contains the Locality. (X.400 Server).

Error: 0B43
Message received when Path not open. This error is severe. (X.400 Server).

Error: 0B44
Incorrect AMOD parameters. This error is severe. (X.400 Server).

Error: 0B45
Message has error in Locality, Product, Index Field that specifies: 1-Source Locality, 4-Destination Locality, 5-Destination Product. This error is severe. User Data 1 contains the Field. User Data 2 contains the Value. (X.400 Server).

Error: 0B46
Entering buffer sequestration. This error is severe. (X.400 Server).

Error: 0B47
Bad Locality ID. This error is severe. (X.400 Server).

Error: 0B48
BUFSTAT has been incremented. This error is severe. User Data 1 contains the Bufstat. (X.400 Server).

Error: 0B49
Wrong length for Locality Data. This error is severe. (X.400 Server).

Error: 0BC0
SAM not found in LRT. This error is fatal. (X.400 Server).

Error: 0BC1
Buffer header pointer (NXTQPTR) corrupt. This error is fatal. (X.400 Server).

Error: 0BC3
Corruption of internal free buffer pool. This error is fatal. (X.400 Server).

Error: 0BC4
Shortage of buffers, BUFSTAT = 4. This error is fatal. (X.400 Server).

Error: 0BC5
Free Ready Pool is empty. This error is fatal. (X.400 Server).

Error: 0BC6
Free Active Pool is empty. This error is fatal. (X.400 Server).

Error: 2100
Message received whilst TRANSPORT is not open. This error is severe. (TRANSPORT).

Error: 2101
Unrecognized event or message. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 2102
Duplicate open product message. This error is severe. (TRANSPORT).

Error: 2103
Spurious open error response. This error is low severity. User Data 1 contains the CB index. (TRANSPORT).

Error: 2104
Spurious close response. This error is low severity. User Data 1 contains the CB index. (TRANSPORT).

Error: 2105
Duplicate open connection request. This error is low severity. User Data 1 contains the CB index. (TRANSPORT).

Error: 2106
Duplicate open connection response. This error is low severity. User Data 1 contains the CB index. (TRANSPORT).

Error: 2107
Illegal Close response. This error is low severity. User Data 1 contains the CB index. (TRANSPORT).

Error: 2108
No LPI credit to receive. This error is severe. (TRANSPORT).

Error: 2109
Primitive has too many elements. This error is severe. (TRANSPORT).

Error: 210A
Closed response received on a closed LPI connection. This error is low severity. (TRANSPORT).

Error: 210B
Passenger message received while pending close response. This error is low severity. (TRANSPORT).

Error: 2110
Invalid (out of range, or not in use) destination on message. This error is severe. (TRANSPORT).

Error: 2111
Incorrect (mismatch) destination on message. This error is severe. (TRANSPORT).

Error: 2112
Message from a lost locality. This error is severe. (TRANSPORT).

Error: 2130
Bad open request received. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 2131
Bad close request received. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 2132
Unknown called TSAP address on a TCON request. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 2133
Unknown calling TSAP address on a TCON request. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 2134
Too many bytes of user data on a primitive. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 2135
Illegal attempt to select expedited data. This error is severe. User Data 1 contains the CB index. . (TRANSPORT).

Error: 2136
Illegal attempt to select a non zero QOS. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 2137
Illegal attempt to request receipt confirmation. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 2138
Bad originator on NDIS indication. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 2139
Unknown called NSAP address on an NCON indication. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 213A
Unknown calling NSAP address on an NCON indication. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 213C
Remote NSAP address too long on an NCON indication. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 213D
Unknown responding TSAP address on a TCON response. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 2150
Bad state to receive a TDAT request. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 2151
Bad state to receive a TEXP request. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 2152
TEXP request received but expedited data not negotiated. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 2153
TCON response received whilst in open, or awaiting an open response, state. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 2154
TCON response received whilst in closed, or awaiting a close response, state. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 2155
TCON response received with user data but in Class 0. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 2156
Received NCON confirmation but NC not waiting for NCON confirmation. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 2157
Bad TC state to receive an NCON confirmation. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 2158
Received NCON confirmation but TC (Class 0 or 2) not in wait for NCON confirmation state. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 2159
TCON response received with user data but in Class 0. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 2180
TPDU header length invalid (larger than 255 bytes). This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 2181
TPDU header length invalid (larger than the remaining space in the NSDU. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 2182
TPDU header length invalid (smaller than 3 bytes). This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 2183
Bad state for a CC TPDU as spotted by the Action Manager. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 2184
CC TPDU received from initiator, *OR* duplicate CC received on a non class 4 connection as spotted by the Action Manager. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 2185
CC TPDU received from initiator, *OR* duplicate CC received on a non class 4 connection as spotted by the Analyser. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 2186
DR TPDU received on Class 0 or 2 connection in WFTRESP. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 2187
Invalid state to receive a DC TPDU. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 2188
Invalid state to receive a DT TPDU. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 2189
DT TPDU has too many elements for a TSDU. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 218A
Invalid state to receive an AK TPDU. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 218B
Arrival of an EA/ED TPDU when expedited data has not been selected. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 218C
Invalid state to receive an ED TPDU. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 218D
Arrival of an unexpected EA TPDU on a Class 2 connection. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 218E
Invalid state to receive an EA TPDU. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 218F
Arrival of an ER TPDU whilst awaiting a TCON response. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 2194
Arrival of a TPDU for a lost TS user. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 2195
RJ TPDU received by TRANSPORT V1.0. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 2196
Arrival of an unrecognized TPDU. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 2197
Bad destination reference on a TPDU. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 2198
Illegal attempt to split a TC onto an NC. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 2199
Inability of TRANSPORT to make a legal split due to a lack of resources. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 219A
Bad sequence number on a DT TPDU. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 219B
EOT badly set on an ED or DT TPDU. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 219C
Checksum failed on received TPDU. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 219D
TPDU header is longer than stated. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 219E
TPDU contains an illegal or unrecognized parameter. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 219F
TPDU size is larger than the negotiated maximum. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 21A0
Length of user data is illegal. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 21A1
Arrival of an illegal AK TPDU on a Class 0 connection. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 21A2
Arrival of an AK TPDU that makes an illegal window change. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 21A3
Arrival of a Class 4 out of sequence AK TPDU. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 21A4
Header too short for the fixed part of a TPDU. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 21A5
Invalid destination reference on a DC TPDU. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 21A6
Arrival of an illegal DC TPDU on a Class 0 connection. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 21A7
Invalid references on a DR TPDU. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 21A8
Arrival of a TPDU that cannot be associated. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 21A9
Bad attempt at class negotiation on a CC TPDU. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 21AA
Bad option parameter on a CC TPDU. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 21AB
Arrival of a bad duplicate CC. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 21AD
Bad Value for the credit to send on a CC TPDU. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 21AE
Bad parameter on a CC TPDU. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 21AF
Arrival of an illegal duplicate CR on a non class 4 connection. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 21B0
Bad parameter on a CR TPDU. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 21B1
Arrival of a CR with a bad source reference. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 21C0
Illegal LPI-STATE combination. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 21C1
Connection affected by buffer shortage. This error is severe. (TRANSPORT).

Error: 21C2
Connection affected by lost locality. This error is severe. (TRANSPORT).

Error: 21C3
Reserved NC has a corrupt multiplexing array. This error is severe. (TRANSPORT).

Error: 21C4
Too many Control Blocks requested at initialization. This error is fatal. (TRANSPORT).

Error: 21C5
Too few Control Blocks requested at initialization. This error is fatal. (TRANSPORT).

Error: 21C6
Bad enqueue detected by TRANSPORT dispatcher. This error is fatal. User Data 1 contains the Component. (TRANSPORT).

Error: 21C7
Splitting array is too small for the requested TC. This error is severe. (TRANSPORT).

Error: 21C8
Source TSAP is too large to fit into the CR TPDU. This error is severe. User Data 1 contains the Source TSAP. User Data 2 contains the identifier. (TRANSPORT).

Error: 21C9
Buffer at global OUTPTR at termination. This error is fatal. (TRANSPORT).

Error: 21CA
Buffer on the retry list at termination. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 21CB
Buffer at a TSDU being built termination. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 21CC
Buffer on the store chain at termination. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 21CD
Buffer at the last ED TPDU sent at termination. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 21CE
Buffer at the open retry pointer at termination. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 21CF
Buffer on the in queue at termination. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 21D0
Buffer on the out queue at termination. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 21D1
Buffer at the build pointer at termination. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 21D2
Buffer at the process pointer at termination. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 21D3
Bad enqueue parameter to O4PIMAIN. This error is severe. (TRANSPORT).

Error: 21D4

The Interface-Manager-Out found no message to send. This error is severe. User Data 1 contains the CB index. (TRANSPORT).

Error: 21E0

TPDU size specified in SCT is too large. This error is fatal. (TRANSPORT).

Error: 2201

A local user who is already logged on has tried to log on again. This error is severe. User Data 1 contains the OPHDR.OPERR1 (1). User Data 2 contains the User Agent index on incoming message. Message Transfer Layer will reject the message (Message Transfer Layer).

Error: 2202

The local user is not configured in the Routing and Directory Table. This error is severe. User Data 1 contains the OPHDR.OPERR1 (2). User Data 2 contains the User Agent index on incoming message. Message Transfer Layer will reject the message (Message Transfer Layer).

Error: 2203

Message Transfer Layer has insufficient UA control blocks to service a received login request. This error is severe. User Data 1 contains the OPHDR.OPERR1 (3). User Data 2 contains the User Agent index on incoming message. Message Transfer Layer will reject the login request. The User Agent should retry the login request (Message Transfer Layer).

Error: 2204

Message Transfer Layer has received an open request while reconfiguration is in progress. This error is severe. User Data 1 contains the OPHDR.OPERR1 (4). User Data 2 contains the User Agent index on incoming message. Message Transfer Layer will reject the message (Message Transfer Layer).

Error: 2205

Message Transfer Layer has received an open request while termination is in progress. This error is severe. User Data 1 contains the OPHDR.OPERR1 (5). User Data 2 contains the User Agent index on incoming message. Message Transfer Layer will reject the message (Message Transfer Layer).

Error: 2206

Message Transfer Layer has received a passenger request while reconfiguration is in progress. This error is severe. User Data 1 contains the Passenger message type. User Data 2 contains the User Agent index on incoming message. Message Transfer Layer will reject the message (Message Transfer Layer).

Error: 2207

The cooperating User Agent has rejected a message because it contains errors. This error is severe. Message Transfer Layer will reject the message (Message Transfer Layer).

Error: 2208

Message Transfer Layer has failed to secure a message on to its internal queues. This

error is severe. Message Transfer Layer will reject the message (Message Transfer Layer).

Error: 2209

An invalid message type has been received. This error is severe. User Data 2 contains the Message type. Message Transfer Layer will discard the message (Message Transfer Layer).

Error: 220A

A message has been received with invalid source LPI address fields. This error is severe. User Data 2 contains the Message type. Message Transfer Layer will discard the message (Message Transfer Layer).

Error: 220B

A message has been received with an invalid destination I address field. This error is severe. User Data 1 contains the Destination I address field. User Data 2 contains the Message type. Message Transfer Layer will discard the message (Message Transfer Layer).

Error: 220C

Message Transfer Layer has received a passenger request when it expects a close. This error is severe. User Data 1 contains the Passenger message type. User Data 2 contains the User Agent index on incoming message. Message Transfer Layer will discard the message (Message Transfer Layer).

Error: 220D

Incoming message does not conform to X400 protocols. This error is severe. User Data 1 contains the State of LPI connection. User Data 2 contains the User Agent control block index. Message Transfer Layer will discard the message (Message Transfer Layer).

Error: 220E

A local user has tried to login using a Logical Name that is reserved for internal use by Message Transfer Layer. This error is severe. User Data 1 contains the OPHDR.OPERR1 (2). User Data 2 contains the User Agent index on incoming message. Message Transfer Layer will reject the message (Message Transfer Layer).

Error: 2210

A syntax error was detected when parsing an MPDU. This error is severe. User Data 1 contains the Syntax Error code: invalid X409 type byte (1), invalid X409 length byte (2), indefinate length primitive (3), too many bytes at this level (4), mandatory item missing (6), unchosen Choice (7), choice chosen twice (8), unexpected type byte (9), repeated item in set (10), originator O/R name unrecognised (13), loop Detected for Delivery Report MPDU (14), P1 contents too long (15), P1 contents too short (16), syntax Error in P2 (17), missing descriptor for contents (19), missing start of sequence (20), read past end of descriptors (21). user Data 2 contains the Syntax Machine Environment Control Block. (Message Transfer Layer).

Error: 2211

A semantic error was detected when parsing an MPDU. This error is severe. User Data 1 contains the Semantic Error Code: error in content type (4), illegal presence of trace information (5), error in trace information (6), loop detected for user or probe

MPDU (7), user data 2 contains the syntax machine environment control block. Message Transfer Layer generates a delivery report MPDU. (Message Transfer Layer).

Error: 2212

Message Transfer Layer is generating a nondelivery report for a specific recipient of the subject message. This error is low priority. User Data 1 contains the index of the recipient in the subject message, User Data 2 contains the nondelivery reason and diagnostic codes for the recipient, coded as: 128 + 16*(Reason Code) + Diagnostic Code. (Message Transfer Layer).

Error: 2220

A bad concatenation of SPDUs has been received. This error is severe. User Data 1 contains the concatenation number. User Data 2 contains the SPDU identifier. Message Transfer Layer sends an S-P-ABORT.ind and an ABORT SPDU. The appropriate recovery action should be taken by the entity that initially opened the session. (Message Transfer Layer).

Error: 2221

An unexpected close (TSI) request has been received. This error is low severity. User Data 1 contains the TS-State: open (1), pending open response (2), pending.close response (3), closed (4). User Data 2 contains the index of source. Message Transfer Layer discards the message. (Message Transfer Layer).

Error: 2222

An unexpected close (TSI) response has been received. This error is low severity. User Data 1 contains the TS-State: open (1), pending open response (2), pending close response (3), closed (4). User Data 2 contains the Index of source. Message Transfer Layer discards the message. (Message Transfer Layer).

Error: 2223

An unexpected passenger message has been received. This error is low severity. User Data 1 contains the TS-State: open (1), pending open response (2), pending close response (3), closed (4). User Data 2 contains the Index of source. Message Transfer Layer discards the message. (Message Transfer Layer).

Error: 2224

Message Transfer Layer has sent an open (TS) error-response. This error is low severity. User Data 1 contains the error code: no free control blocks (3827). User Data 2 contains the Index of source. (Message Transfer Layer).

Error: 2225

Unexpected message received across TSI. This error is low severity. User Data 1 contains the error in destination (1), error in source address or session control block state (2). User Data 2 contains the buffer header message type field. Message Transfer Layer discards the message (Message Transfer Layer).

Error: 2226

Unexpected message received across SSI. This error is low severity. User Data 1 contains the error in destination (1), error in source address or session control block state (2). User Data 2 contains the buffer header message type field. Message Transfer Layer discards the message (Message Transfer Layer).

Error: 2227

An unexpected open (TSI) response has been received. This error is low severity. User Data 1 contains the TS-State: open (1), pending open response (2), pending. close response (3), closed (4). User Data 2 contains the Index of source. Message Transfer Layer discards the message. (Message Transfer Layer).

Error: 2228

Repeated open (TS) request has been received. This error is low severity. User Data 2 contains the Index of source. Message Transfer Layer discards the message (Message Transfer Layer).

Error: 2229

Error detected in SPDU. This error is severe. User Data 1 contains the format error (1), SPDU-id error (2). User Data 2 contains the SPDU identifer. Message Transfer Layer sends an S-P-ABORT.ind, and an ABORT SPDU. The appropriate recovery action should be taken by the entity that initially opened the session. (Message Transfer Layer).

Error: 222A

Protocol Error due to a received session primitive. This error is severe. User Data 1 contains the session state. User Data 2 contains the Session primitive type. Message Transfer Layer sends an S-P-ABORT.ind, and an ABORT SPDU. The appropriate recovery action should be taken by the entity that initially opened the session. (Message Transfer Layer).

Error: 222B

Message Transfer Layer has received an open (TS) error-response. This error is severe. User Data 1 contains the error code: product not supported by SAM (2560), product not found in SCT (2561), resource identifier not recognised (2569), unrecognised open product (2624), no free CBs at transport layer (3827). User Data 2 contains the Index of source. For the first four error codes the half-opened session connection is aborted, otherwise the open session is retried. (Message Transfer Layer).

Error: 222C

Protocol Error due to a received SPDU. This error is severe. User Data 1 contains the session state. User Data 2 contains the SPDU identifer. Message Transfer Layer sends an S-P-ABORT.ind, and an ABORT SPDU. The appropriate recovery action should be taken by the entity that initially opened the session. (Message Transfer Layer).

Error: 222D

A TSDU has arrived at Message Transfer Layer, whose length is greater than the negotiated maximum TSDU size. This error is severe. User Data 1 contains the TSDU length. User Data 2 contains the negotiated maximum TSDU size. Message Transfer Layer sends an S-P-ABORT.ind, and an ABORT SPDU. The appropriate recovery action should be taken by the entity that initially opened the session. (Message Transfer Layer).

Error: 222E

Error detected in session primitive. This error is severe. User Data 1 contains the Format error (1), SPDU-id error (2). User Data 2 contains the primitive type. Message Transfer Layer sends an S-P-ABORT.ind, and an ABORT SPDU. The appropriate

recovery action should be taken by the entity that initially opened the session. (Message Transfer Layer).

Error: 2231

An S-connect indication has been received that has no SS-User data. This error is severe. User Data 1 contains the session control block index. Message Transfer Layer will reject the indication (Message Transfer Layer).

Error: 2232

An S-connect indication for a new connection has been refused because there are no free RTS control blocks. This error is severe. Message Transfer Layer will reject the indication (Message Transfer Layer).

Error: 2233

An S-connect indication has been rejected because the the SS-user data is invalid. This error is severe. User Data 1 contains the current offset in SS-User data. Message Transfer Layer will reject the indication (Message Transfer Layer).

Error: 2234

An S-connect indication for a recovery has been refused because the RTS was unable to validate the session connection identifier. This error is severe. Message Transfer Layer will reject the indication (Message Transfer Layer).

Error: 2235

An exception report has been received for the current activity. This error is low severity. User Data 1 contains the action taken by Message Transfer Layer: 0 = discard current activity, 1 = interrupt current activity. User Data 2 contains the reason code on exception report: 0 = non-specific error, 1 = receiving ability jeopardised, 3 = sequence error, 5 = local SS-user error, 6 = unrecoverable procedure error, (Message Transfer Layer).

Error: 2236

Message Transfer Layer has detected a sequence error on the last received checkpoint. This error is low severity. User Data 1 contains the action taken by Message Transfer Layer: 0 = discard current activity, 1 = interrupt current activity (Message Transfer Layer).

Error: 2240

The Remote MTA has rejected an attempt to open an association. This error is severe. User Data 1 contains the association control block index. User Data 2 contains the reason code. If the reason code specifies that the attempt should be retried (eg. busy), Message Transfer Layer will try again. Otherwise, the attempt will be aborted. (Message Transfer Layer).

Error: 2241

Message Transfer Layer has received a message containing an invalid destination index address field. This error is severe. User Data 1 contains the association control block index. User Data 2 contains the message identifier. Message Transfer Layer will discard the message (Message Transfer Layer).

Error: 2245

Message Transfer Layer has received an ROPEN indication that contains an error. This error is severe. User Data 1 contains the error code: 0 = busy, 1 = cannot recover, 2 = authentication failure, 3 = unacceptable dialogue mode, 4 = invalid SSAP. User Data 2 contains therRemote MTA control block index. Message Transfer Layer builds a negative ROPEN response (Message Transfer Layer).

Error: 2246

Message Transfer Layer has received an ROPEN confirmation whose RTS user data has failed the validation test. This error is severe. User Data 1 contains the association control block index. User Data 2 contains the remote MTA control block index. Message Transfer Layer closes the association (Message Transfer Layer).

Error: 2250

The gateway component has received an invalid message. This error is severe. User Data 2 contains the message type. Message Transfer Layer discards the message (Message Transfer Layer).

Error: 2251

The gateway component has received a message with an invalid source LPI. User Data 2 contains the message type. Message Transfer Layer discards the message (Message Transfer Layer).

Error: 2252

The gateway component has received a message with an invalid destination I. User Data 1 contains the destination I. User Data 2 contains the message type. Message Transfer Layer discards the message (Message Transfer Layer).

Error: 2253

The gateway component has received a close request. User Data 1 contains the gateway_connection_state. User Data 2 contains the gateway_connection cb index. Message Transfer Layer discards the message (Message Transfer Layer).

Error: 2254

The gateway component has received a close response. User Data 1 contains the gateway_connection_state. User Data 2 contains the gateway_connection cb index. Message Transfer Layer discards the message (Message Transfer Layer).

Error: 2255

The gateway component has received a passenger request in an invalid state. This error is severe. User Data 1 contains the gateway_connection_state. User Data 2 contains the gateway_connection cb index. Message Transfer Layer discards the message (Message Transfer Layer).

Error: 2256

The gateway component has received a passenger. User Data 1 contains the gateway_connection_state. User Data 2 contains the gateway_connection cb index. Message Transfer Layer discards the message (Message Transfer Layer).

Error: 2257

The gateway component has received a receive. User Data 1 contains the

gateway_connection_state. User Data 2 contains the gateway_connection cb index. Message Transfer Layer discards the message (Message Transfer Layer).

Error: 2258

Message Transfer Layer is unable to generate an alternative name for a body file passed across the gateway interface in a data request. This error is severe. User Data 1 contains the gateway_connection cb index. Message Transfer Layer builds, and sends, the gateway a negative data confirm message (Message Transfer Layer).

Error: 2259

Message Transfer Layer fails in its attempt to rename a body file passed across the gateway interface in a data request. This error is severe. User Data 1 contains the gateway_connection cb index. User Data 2 contains the reason code returned by the object manager. Message Transfer Layer builds, and sends, the gateway a negative data confirm message (Message Transfer Layer).

Error: 225A

Message Transfer Layer cannot accept the connection requested across the gateway interface. This error is low severity. User Data 1 contains the OPHDR.OPERR1 field: user not configured in R&D table (2), no free control blocks (3), Message Transfer Layer reconfiguring (4), Message Transfer Layer terminated (5). User Data 2 contains the source index of open request. Message Transfer Layer rejects the open request (Message Transfer Layer).

Error: 225B

Message Transfer Layer cannot accept a passenger request. This error is low severity. User Data 1 contains the OPHDR.OPERR1 field: user not configured in R&D table (2), no free control blocks (3), Message Transfer Layer reconfiguring (4), Message Transfer Layer terminated (5). User Data 2 contains the source index of passenger request. Message Transfer Layer rejects the passenger request (Message Transfer Layer).

Error: 225C

The gateway component has received an invalid passenger message. This error is low severity. User Data 1 contains the passenger message type. User Data 2 contains the gateway connection cb. Message Transfer Layer discards the message (Message Transfer Layer).

Error: 2290

In the delay between delivering a message to an alternate recipient, and generating the delivery report, the alternate recipient has been deconfigured. The identity of the alternate recipient has therefore been lost, and no DR can be generated. This error is low severity. User Data 1 contains the logical name of deconfigured recipient. Message Transfer Layer will abort the generation of the DR (Message Transfer Layer).

Error: 2291

In the delay between delivering a message to an alternate recipient, and generating the delivery report, the alternate recipient has been deconfigured, and non-UA has been put in its place. The identity of the alternate recipient has therefore been lost, and no DR can be generated. This error is low severity. User Data 1 contains the logical nName of deconfigured recipient. User Data 2 contains the index into R&D table of recipient. Message Transfer Layer will abort the generation of the DR (Message Transfer Layer).

Error: 22A0

Error opening read file. This error is low severity. User Data 1 contains the logical name. User Data 2 contains the queue manager operation. The queue manager suspends all operations on the affected queue. (Message Transfer Layer).

Error: 22A1

Read failure. This error is low severity. User Data 1 contains the logical name. User Data 2 contains the queue manager operation. The queue manager suspends all operations on the affected queue. (Message Transfer Layer).

Error: 22A2

Error deleting read file. This error is low severity. User Data 1 contains the logical name. User Data 2 contains the queue manager operation. The queue manager suspends all operations on the affected queue. (Message Transfer Layer).

Error: 22A3

Error closing read file. This error is low severity. User Data 1 contains the logical name. User Data 2 contains the queue manager operation. The queue manager suspends all operations on the affected queue. (Message Transfer Layer).

Error: 22A4

Error opening write file. This error is low severity. User Data 1 contains the logical name. User Data 2 contains the queue manager operation. The queue manager suspends all operations on the affected queue. (Message Transfer Layer).

Error: 22A5

Write failure. This error is low severity. User Data 1 contains the logical name. User Data 2 contains the queue manager operation. The queue manager suspends all operations on the affected queue. (Message Transfer Layer).

Error: 22A6

Error deleting read file. This error is low severity. User Data 1 contains the logical name. User Data 2 contains the queue manager operation. The queue manager suspends all operations on the affected queue. (Message Transfer Layer).

Error: 22A7

Error closing read file. This error is low severity. User Data 1 contains the logical name. User Data 2 contains the queue manager operation. The queue manager suspends all operations on the affected queue. (Message Transfer Layer).

Error: 22A8

Error in queue directory. This error is severe. User Data 1 contains the logical name. User Data 2 contains the queue manager operation. If this error occurs during initialization, Message Transfer Layer terminates; otherwise all queue manager establish and disestablish operations are inhibited. (Message Transfer Layer).

Error: 22A9

Queue activity suspended after previous error. This error is low severity. User Data 1 contains the logical name. User Data 2 contains the queue manager operation. (Message Transfer Layer).

Error: 22AA

Error in data read from queue or queue directory. This error is low severity. User Data 1 contains the logical name. User Data 2 contains the queue manager operation. The queue manager suspends all operations on the affected queue. (Message Transfer Layer).

Error: 22BB

Message Transfer Layer has been unable to open the file containing the routing and directory table. This error is fatal. Message Transfer Layer terminates (Message Transfer Layer).

Error: 22BC

The routing and directory table contains a logical name whose value exceeds the upper bound of the queue directory. This error is fatal. Message Transfer Layer terminates (Message Transfer Layer).

Error: 22BD

The routing and directory table contains a record whose index is greater than the maximum configured. This error is fatal. Message Transfer Layer terminates (Message Transfer Layer).

Error: 22BF

Message Transfer Layer was unable to find the Journal file. This error is severe. Message Transfer Layer inhibits logging to the journal (Message Transfer Layer).

Error: 22D1 2290

The number of UA control blocks in the SCT exceeds the number specified at compile time. This error is fatal. User Data 1 contains the SCT value. User Data 2 contains the compile time value. Either reduce the SCT value, or recompile with more control blocks (Message Transfer Layer).

Error: 22D2 2291

The number of MTA control blocks in the SCT exceeds the number specified at compile time. This error is fatal. User Data 1 contains the SCT value. User Data 2 contains the compile time value. Either reduce the SCT value, or recompile with more control blocks (Message Transfer Layer).

Error: 22D3 2292

The number of ASS control blocks in the SCT exceeds the number specified at compile time. This error is fatal. User Data 1 contains the SCT value. User Data 2 contains the compile time value. Either reduce the SCT value, or recompile with more control blocks (Message Transfer Layer).

Error: 22D4 2293

The number of RTS control blocks in the SCT exceeds the number specified at compile time. This error is fatal. User Data 1 contains the SCT value. User Data 2 contains the compile time value. Either reduce the SCT value, or recompile with more control blocks (Message Transfer Layer).

Error: 22D5 2294

The number of SS control blocks in the SCT exceeds the number specified at compile

time. This error is fatal. User Data 1 contains the SCT value. User Data 2 contains the compile time value. Either reduce the SCT value, or recompile with more control blocks (Message Transfer Layer).

Error: 22D6

The number of FM control blocks in the SCT exceeds the number specified at compile time. This error is fatal. User Data 1 contains the SCT value. User Data 2 contains the compile time value. Either reduce the SCT value, or recompile with more control blocks (Message Transfer Layer).

Error: 22D7

The number of OM control blocks in the SCT exceeds the number specified at compile time. This error is fatal. User Data 1 contains the SCT value. User Data 2 contains the compile time value. Either reduce the SCT value, or recompile with more control blocks (Message Transfer Layer).

Error: 22D8

The number of QM control blocks in the SCT exceeds the number specified at compile time. This error is fatal. User Data 1 contains the SCT value. User Data 2 contains the compile time value. Either reduce the SCT value, or recompile with more control blocks (Message Transfer Layer).

Error: 22D9

The number of MD control blocks in the SCT exceeds the number specified at compile time. This error is fatal. User Data 1 contains the SCT value. User Data 2 contains the compile time value. Either reduce the SCT value, or recompile with more control blocks (Message Transfer Layer).

Error: 22DA

The number of SME control blocks in the SCT exceeds the number specified at compile time. This error is fatal. User Data 1 contains the SCT value. User Data 2 contains the compile time value. Either reduce the SCT value, or recompile with more control blocks (Message Transfer Layer).

Error: 22DB

The number of SME control blocks available is too low to support the required Message Transfer Layer configuration. This error is fatal. User Data 1 contains the number of SME control blocks available. User Data 2 contains the number of SME control blocks required. Add more control blocks (Message Transfer Layer).

Error: 22DC

The number of RTS control blocks available is too low to support the required Message Transfer Layer configuration. This error is fatal. User Data 1 contains the number of RTS control blocks available. User Data 2 contains the number of RTS control blocks required. Add more control blocks (Message Transfer Layer).

Error: 22DD

The number of MD control blocks available is too low to support the required Message Transfer Layer configuration. This error is fatal. User Data 1 contains the number of MD control blocks available. User Data 2 contains the number of MD control blocks required. Add more control blocks (Message Transfer Layer).

22DA Number of Queue directory entries exceeds the max value set at compile time
22DE Number of Gateway entity control blocks exceeds the number specified at ...

Error: ~~22DE~~ 22DF

The number of SS control blocks available is too low to support the required Message Transfer Layer configuration. This error is fatal. User Data 1 contains the number of SS control blocks available. User Data 2 contains the number of SS control blocks required. Add more control blocks (Message Transfer Layer).

Error: ~~22DF~~ 22DB

The number of Routing Table lines requested exceeds the maximum value set at compile-time. This error is low severity. User Data 1 contains the number of lines available. User Data 2 contains the number of lines requested; none are contained unless error 22BD occurs later, in which case change value in SCT (Message Transfer Layer).

Error: 22E0

Error while reading in the R&D table. This error is fatal. Message Transfer Layer will terminate; check the R&D table (Message Transfer Layer).

Error: 22E1

The number of MTAs configured in the R&D table is greater than the number of MTA control blocks configured in the SCT. This error is fatal. User Data 1 contains the number of MTAs in the R&D table. User Data 2 contains the number of MTA control blocks available. Message Transfer Layer will terminate; add more control blocks (Message Transfer Layer).

Error: 22E2

The number of associations configured in the R&D table is greater than the number of ASS control blocks configured in the SCT. This error is fatal. User Data 1 contains the number of associations in the R&D table. User Data 2 contains the number of ASS control blocks available. Message Transfer Layer will terminate; add more control blocks (Message Transfer Layer).

Error: 22E3

The number of gateway entities configured in the R&D table is greater than the number of gateway entity control blocks configured in the SCT. This error is fatal. User Data 1 contains the number of gateway entities in the R&D table. User Data 2 contains the number of gateway entity control blocks available. Message Transfer Layer will terminate; add more control blocks (Message Transfer Layer).

Error: 22E4

The number of gateway connections configured in the R&D table is greater than the number of gateway connection control blocks configured in the SCT. This error is fatal. User Data 1 contains the number of gateway connections in the R&D table. User Data 2 contains the number of gateway connection control blocks available. Message Transfer Layer will terminate; add more control blocks (Message Transfer Layer).

Error: 22E5

An MTA with no associations has been found in the R&D table. This error is fatal. User Data 1 contains the logical name of MTA. Message Transfer Layer will terminate; check the configuration of the MTA (Message Transfer Layer).

Error: 22E6

A gateway entity with no connections has been found in the R&D table. This error is fatal. User Data 1 contains the logical name of gateway. Message Transfer Layer will terminate; check the configuration of the gateway (Message Transfer Layer).

Error: 22E7

An illegal operation has been attempted during an APPEND reconfiguration. This error is severe. User Data 1 contains the logical name. Message Transfer Layer will terminate; check the new R&D table to ensure that no users, adjacent MTAs, or gateways have been deconfigured or moved. (Message Transfer Layer).

Error: 22E8

The R&D table contains multiple local attribute records. This error is fatal. Message Transfer Layer will terminate; check the R&D table (Message Transfer Layer).

Error: 22E9

The R&D table does not contain a local attribute record. This error is fatal. Message Transfer Layer will terminate; check the R&D table (Message Transfer Layer).

Error: 22EA

Message Transfer Layer has detected a file error while updating the version number. This error is fatal. Message Transfer Layer terminates (Message Transfer Layer).

Error: 22EB

A mismatch between the queue directory and the R&D table has been detected by Message Transfer Layer during recovery. This error is fatal. User Data 1 contains the logical name for which the mismatch was found. User Data 2 indicates the mismatch: previously unused logical name now assigned to a Message Transfer Agent (MTA) (2), previously unused logical name now assigned to an User Agent (UA) (3), previously unused logical name now assigned to a gateway (GW) (4), logical name already in use but now assigned to an MTA (5), logical name already in use but now assigned to an UA (6), logical name already in use but now assigned to an GW (7), logical name was in use but is now unassigned (8). Message Transfer Layer will terminate; check the R&D table (Message Transfer Layer).

Error: 22EC

A failure has occured when Message Transfer Layer attempted to rename the R&D table at the end of remote reconfiguration. This error is fatal. User Data 1 contains the object manager error code. Message Transfer Layer terminates (Message Transfer Layer).

Error: 22ED

A message other than an open (product) response or a repeated open request has been received during product initialisation. This error is severe. Message Transfer Layer will discard the message (Message Transfer Layer).

Error: 22EE

A message has been received that specifies an unsupported interface type. This error is severe. Message Transfer Layer will discard the message (Message Transfer Layer).

Error: 22F0

The operator component has received a message with an invalid source LPI. This error is severe. Message Transfer Layer discards the message (Message Transfer Layer).

Error: 22F1

The operator component has received an invalid message type. This error is severe. User Data 1 contains the Message type. Message Transfer Layer discards the message (Message Transfer Layer).

Error: 22F2

The operator component has received an open response message in an invalid state. This error is severe. User Data 1 contains the operator component state. Message Transfer Layer discards the message (Message Transfer Layer).

Error: 22F3

The operator component has received a data message in an invalid state. This error is severe. User Data 1 contains the operator component state. Message Transfer Layer discards the message (Message Transfer Layer).

Error: 22F4

The operator component has received a status message in an invalid state. This error is severe. User Data 1 contains the operator component state. User Data 2 contains the status message type. Message Transfer Layer discards the message (Message Transfer Layer).

Error: 22F5

The operator component has received an open error response. This error is severe. User Data 1 contains 0 (fatal), or 1 (non-fatal). User Data 2 contains the ADMIN_X400 error code. For a fatal error, the operator component goes to an error state and stays there; for a non-fatal error the open is retried. (Message Transfer Layer). The operator component is processing a lost locality notification. This error is severe. User Data 1 contains the locality identifier. Outstanding commands and unsolicited messages are released, and Message Transfer Layer attempts to reopen the connection. (Message Transfer Layer).

Error: 22F7

The operator component is processing a buffer shortage notification. This error is severe. User Data 1 contains the Buffer shortage level. For level 1, queued unsolicited information messages are released; for level 3, all outstanding commands and unsolicited messages are released. (Message Transfer Layer).

Error: 22F8

Reference to invalid entity detected in DISPLAY command. This error is severe. User Data 1 contains the invalid entity. User Data 2 contains the error code: invalid_primary_entity (1), invalid_primary_parameter (2), invalid_secondary_entity (3). Message Transfer Layer responds to the command with an invalid entity message. Retype command with correct entity (Message Transfer Layer).

Error: 22F9

Cannot build unsolicited message. This error is severe. User Data 1 contains the Event type: startup (0), queue (1), error (2), association (3). The logging message does not appear (Message Transfer Layer).

Error: 22FA
The operator component has received an error response from the queue manager. This error is severe. User Data 1 contains the queue manager error code. Message Transfer Layer will abort the generation of the DR (Message Transfer Layer).

Error: 2200
The transport connection has been lost. This error is severe. User Data 1 contains the routine identifier (trace constant). User Data 2 contains the RTS control block index. Message Transfer Layer will enter session recovery mode (Message Transfer Layer).

Error: 222C
A session layer protocol error has been detected, e.g., a primitive collision. This error is severe. User Data 1 contains the routine identifier (trace constant). User Data 2 contains the RTS control block index. Message Transfer Layer will enter session recovery mode (Message Transfer Layer).

Error: 2263
Message Transfer Layer is unable to open the transport connection due to a configuration error. This error is severe. User Data 1 contains the routine identifier (trace constant). User Data 2 contains the RTS control block index. Message Transfer Layer will abort the corresponding association. The R&D table should be checked (Message Transfer Layer).

Error: F001
Could not allocate synchronizer group (isolation layer).

Error: F002
Could not allocate KOS timer (isolation layer).

Error: F003
Could not allocate KOS synchronizer (isolation layer).

Error: F004
Could not add synchronizer to group (isolation layer).

Error: F005
Could not start repetetive timer (isolation layer).

Error: F006
Could not register server (isolation layer).

Error: F007
Could not add session synchronizer to group (isolation layer).

Error: F008
Could not create/modify HLN (isolation layer).

Error: F009
Could not find service for code (isolation layer).

Error: F00A
Error when issuing a session request (isolation layer).

Error: F00B
No handler for session (isolation layer).

Error: F00C
Error when picking up a new session (isolation layer).

Error: F00D
Error when picking up a new event (isolation layer).

Error: F00E
Error when accepting an ISC Session (isolation layer).

Error: F00F
ISC Session rejected (no server?) (isolation layer).

Error: F010
Error when picking up a session response (isolation layer).

Error: F011
No memory for SSCB (isolation layer).

Error: F012
ISC subservice refused session (isolation layer).

Error: F013
Fatal ON error from main() (isolation layer).

Error: F014
Invalid SSCB for this event (isolation layer).

Error: F201
Failed to clear X.25 VC (X$CLR) (Network Service). Primenet status is User Data 1.

F202 call to XL CONN failed

Error: F203
X.25 VC cleared by network or remote MTA. Diagnostic indicates abnormal termination (Network Service).

Error: F204
X.25 VC cleared by local MTA. Diagnostic indicates abnormal termination (Network Service).

Error: F205
Failed to setup X.25 connection (Network Service).

Error: F206
X.25 reset received. (Network Service).

Error: F207
Incoming X.25 call cannot be accepted due to lack of resources (Network Service).

Error: F208
X.25 packet received with Q-bit set or interrupt packet. Possibly caused by connecting to PAD/Remote Login Service. Either configuration problem or (prior to Revision 22) the remote X.400 server is not running. (Network Service).

Error: F209
Failed to accept X.25 call (XLACPT) (Network Service).

Error: F20A
NS-user violated protocol (Network Service).

Error: F20B
Failed to pick up call (XLGC$) (Network Service).

Error: F300
Fatal amod initialization error (Network Service).

Error: F304
Handler array already full (AMOD).

Error: F306
ISC receive error (AMOD).

Error: F307
ISC send Error (AMOD).

Error: F400
The BIOS PMOD has detected an operating system error when performing the current BIOS operation (isolation layer). User Data 1 contains the Primos error code. User Data 2 indicates the operation being performed: open file for read (1), open file for write (2), open file for append (3), close file (4), delete file (5).

Error: F500
Handler Array already full.

Error: F501
Failure during close.

Error: F502
ISC routine failed.

Error: F503
Data area too small for write.

Error: F504
Error whilst sending data.

Error: F505
Error whilst processing a session request from an RI.

Error: F506
Error whilst reading message.

Error: F507
FFD not given to scppwrit.

Error: F601
Failed to open routing cache file.

Error: F602
No T-SAP in template SCT (Bug).

Error: F603
No N-SAP in template SCT (Bug).

Error: F604
No X.400 product data in SCT (Bug).

Error: F605
No R&D table name (Bug).

Error: F606
Syntax error in subnet specification (Bug).

Error: F607
Illegal value in subnet specification (Bug).

Error: F608
Subnet out of range, or not in SCT. Too many subnetworks specified.

Error: F609
The Administrator has tried to configure too many active local users (isolation layer). User Data 1 contains the number of active local users configured by the Administrator. User Data 2 contains the maximum permitted number of active local users. The number of active local users is configured to that value contained in User Data 2.

INDEX

INDEX

E

SURVEYS

READER RESPONSE FORM

Prime X.400 API Development Kit Administrator's Guide
DOC11232-1LA

Your feedback will help us continue to improve the quality, accuracy, and organization of our publications.

1. How do you rate this document for overall usefulness?

   ☐ *excellent* ☐ *very good* ☐ *good* ☐ *fair* ☐ *poor*

2. What features of this manual did you find most useful?

   _____
   _____
   _____
   _____
   _____
   _____

3. What faults or errors in this manual gave you problems?

   _____
   _____
   _____
   _____
   _____
   _____

4. How does this manual compare to equivalent manuals produced by other computer companies?

   ☐ *Much better* ☐ *Slightly better* ☐ *About the same*
   ☐ *Much worse* ☐ *Slightly worse* ☐ *Can't judge*

5. Which other companies' manuals have you read?

   _____
   _____

Name:_____ Position:_____

Company:_____

Address:_____

_____

_____ Postal Code:_____

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

First Class Permit #531 Natick, Massachusetts 01760

BUSINESS REPLY MAIL

Postage will be paid by:

Prime™

Attention: Technical Publications
Bldg 10
Prime Park, Natick, Ma. 01760