HP MAILbus 400 Message Transfer Agent

Planning and Setup

October 2006

Revision/Update information:	This is a revised manual.
Operating System Version:	OpenVMS Alpha Version 7.3-2 and above OpenVMS VAX Version 7.3
Revision/Update Information:	HP MAILbus 400 Message Transfer Agent Version 3.2-12 for OpenVMS

Hewlett-Packard Company
Palo Alto, California

© Copyright 2006 Hewlett-Packard Development Company, L.P.

Confidential computer software. Valid license from HP required for possession, use, or copying. Consistent with FAR 12.211 and 12.212, Commercial Computer Software, Computer Software Documentation, and Technical Data for Commercial Items are licensed to the U.S. Government under vendor's standard commercial license.

The information contained herein is subject to change without notice. The only warranties for HP products and services are set forth in the express warranty statements accompanying such products and services. Nothing herein should be construed as constituting an additional warranty. HP shall not be liable for technical or editorial errors or omissions contained herein.

cc:Mail is a registered trademark of cc:Mail, Inc.

ISOCOR and ISOGATE are trademarks of ISOCOR.

ISOPLEX 800 is a registered trademark of ISOCOR.

Lotus Notes is a registered trademark of Lotus Development Corp.

Microsoft is a are registered trademark of Microsoft Corporation.

NetWare is a registered trademark of Novell, Inc.

OpenServer 400 is a trademark of Retix.

OSI is a registered trademark of CA Management, Inc.

Retix is a registered trademark of Retix.

UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Ltd.

Printed in the US

The HP OpenVMS documentation set is available on CD-ROM.

This document was prepared using DECdocument, Version 3.3-1B.

Contents

Preface xiii

Part I Introduction

1 The HP MAILbus 400 Message Transfer Agent

1.1 What is the MAILbus 400 MTA? 1–1
1.1.1 The 1992 MHS Standards 1–2
1.2 The MAILbus 400 MTA and the HP Enterprise Directory Service 1–3
1.3 The MAILbus 400 MTA and MAILbus 1–4
1.4 Operating Over DECnet-Plus and TCP/IP Networks 1–5
1.4.1 The MAILbus 400 MTA and HP DECnet-Plus 1–6
1.4.2 The MAILbus 400 MTA and TCP/IP 1–6
1.5 Conformance to the Enterprise Management Architecture 1–6
1.6 What to Do Next? 1–9

2 An X.400 Message Handling System and its Components

2.1 The Role of the Message Transfer Agent 2–2
2.2 The Role of the User Agent 2–3
2.3 The Role of the Access Unit 2–3
2.4 The Role of the Message Store 2–3
2.5 Messages, Reports and Probes in an X.400 MHS 2–4
2.5.1 What is a Message? 2–4
2.5.1.1 Structure of a Message 2–4
2.5.1.2 Interpersonal Messages 2–5
2.5.2 What is a Report? 2–6
2.5.3 What is a Probe? 2–7
2.6 What to do Next 2–7

Part II Planning

3 Planning a Naming Scheme

3.1 Defining a Hierarchy for Your Organization 3–1
3.2 Choosing a Naming Scheme 3–3

4 Naming Routing Domains, MTAs and Agents

4.1 Routing Domains 4–1
4.1.1 Naming Your Routing Domain 4–3
4.1.2 Global Domain Identifier 4–4
4.2 Understanding the Layout of Your Routing Domain 4–5
4.2.1 MTA Sets 4–5
4.2.2 Dividing Your Routing Domain into Areas 4–7
4.3 Naming MTAs 4–9
4.4 Naming Agents 4–11
4.4.1 Registered Agents 4–12
4.4.1.1 XAPI Interface 4–12
4.4.1.2 Shared File Interface 4–12
4.4.2 Unregistered Agents 4–13
4.5 Naming Other Routing Domains 4–13
4.6 Peer MTA Entities 4–16
4.6.1 Automatically-Configured Peer MTA Entities 4–17
4.6.2 Manually-Configured Peer MTA Entities 4–17
4.7 Summary of Planning MTA and Routing Domain Names 4–20
4.8 Summary of Planning Peer MTA Entities 4–22
4.9 Summary of Planning Agent Entities 4–26
4.10 What to Do Next 4–27

5 O/R Addresses and Routing Information

5.1 Addressing in an X.400 MHS 5–1
5.1.1 O/R Addresses 5–2
5.1.1.1 Mnemonic O/R Addresses 5–2
5.1.1.2 Numeric O/R Addresses 5–3
5.1.1.3 Postal O/R Addresses 5–3
5.1.1.4 Terminal O/R Addresses 5–3
5.1.2 Ensuring the Uniqueness of O/R Addresses 5–3
5.2 Why You Need O/R Addresses 5–5
5.3 O/R Address Attributes 5–7

5.3.1 O/R Address Attribute Descriptions 5–10
5.3.1.1 Country Name 5–10
5.3.1.2 Administration Domain Name 5–11
5.3.1.3 Private Domain Name 5–12
5.3.1.4 Organization Name 5–13
5.3.1.5 Organizational Unit Name 5–13
5.3.1.6 Common Name 5–14
5.3.1.7 Personal Name 5–14
5.3.1.8 Domain Defined 5–14
5.3.1.9 Numeric User Identifier 5–15
5.4 Allocating O/R Address Attributes to Individuals in Your Routing Domain 5–15
5.4.1 Resolving Naming Ambiguities 5–16
5.5 Representing O/R Addresses for Individuals in Other Routing Domains 5–17
5.5.1 Complete or Partial O/R Addresses 5–17
5.6 Alias O/R Addresses 5–18
5.7 Routing Information for Complete O/R Addresses 5–19
5.8 Routing Information for Partial O/R Addresses 5–22
5.9 Routing Information for Other Routing Domains 5–25
5.10 Routing Information for Different Forms of O/R Address 5–26
5.11 Content Information 5–27
5.12 Distribution Lists 5–27
5.13 Permission to Send Mail to Another X.400 Management Domain 5–28
5.14 Warning Text on Messages Arriving from Other X.400 Management Domains 5–29
5.15 Recovering Messages from an MTA/recover_msg 5–29
5.16 What to Do Next 5–30

6 Planning the Use of the HP Enterprise Directory Service

6.1 The MTA and the HP Enterprise Directory Service 6–1
6.2 Regulating Access to the Directory 6–2
6.3 HP Enterprise Directory Service Setup Required 6–2
6.4 Ensuring Routing Information is Available to MTAs 6–3

7 Summary of Managing Routing Information

7.1 The MTS Module 7–1
7.1.1 Replication 7–2
7.1.2 Access Control 7–3
7.2 Routing Information Held Locally at an MTA 7–4
7.3 Routing Information Held in the Directory 7–5
7.4 O/R Address Information 7–6
7.5 Routing Domain Information 7–11
7.6 Area and MTA Information 7–15
7.7 Foreign Address Information 7–16
7.8 What to Do Next 7–19

Part III Setup

8 Setting Up Your Routing Domain on a Tru64 UNIX System

8.1 Tasks to Complete Before Setting Up any MTAs 8–2
8.1.1 Ensure the Directory Server Subset is Installed 8–2
8.1.2 Ensure the Directory Base Subset is Installed 8–3
8.1.3 Ensure the Required MTA Subsets are Installed 8–3
8.1.4 What to Do Next 8–3
8.2 Upgrading from a Previous Version of the Product 8–3
8.2.1 Check That There is a DUA Defaults File on the System . . . 8–4
8.2.2 Add Your Customizations to the New Startup File 8–4
8.2.3 Run the MTA Setup Procedure 8–4
8.2.4 Create the MTA Entry in the Directory 8–6
8.2.5 Start the MTA 8–6
8.2.6 Checking create_mta_cons_templates.ncl 8–6
8.2.7 What to Do Next 8–7
8.3 Setting Up an MTA in the Routing Domain for the First Time 8–7
8.3.1 Set Up Access to a DSA 8–7
8.3.1.1 Configure the DSA 8–7
8.3.1.2 Create a Naming Context for the MTA 8–8
8.3.1.3 Create the DUA Defaults File 8–8
8.3.2 Create the Routing Domain and Area Entries for the MTA 8–9
8.3.2.1 Create the MTS Entity for Your Routing Domain 8–9
8.3.2.2 Add Global Domain Identifiers 8–10
8.3.2.3 Create Area Entries 8–11
8.3.3 What to Do Next 8–11
8.4 Adding More MTAs to Your Routing Domain 8–11

8.4.1 Set Up Access to a DSA 8–12
8.4.2 Provide Authorization to Manage the MTA's Routing Information 8–13
8.4.3 Run the MTA Setup Procedure 8–13
8.4.3.1 Create the MTA Entry in the Directory 8–15
8.4.4 Start the MTA 8–15
8.4.5 Checking create_mta_cons_templates.ncl 8–15
8.5 Using the MTA Verification Procedure 8–16
8.6 Setting Up an MTA to Contact a Shadow DSA 8–17
8.7 Keeping the MTA Startup Script Up-to-Date 8–18
8.8 What to Do Next 8–19
8.9 Setting Up Remote Access to the Routing Information in the Directory 8–19

9 Setting Up Your Routing Domain on an OpenVMS System

9.1 Setting Up the MTA in a Cluster 9–2
9.2 Tasks to Complete Before Setting Up Any MTAs 9–3
9.2.1 Ensure the Directory Server Component is Installed 9–3
9.2.2 Ensure the Directory Base Component is Installed 9–3
9.2.3 Ensure the Required MTA Components are Installed 9–3
9.2.4 What to Do Next 9–3
9.3 Upgrading from a Previous Version of the Product 9–4
9.3.1 Check That There is a DUA Defaults File on the System . . . 9–4
9.3.2 Initialize the MTS Process 9–4
9.3.3 Run the MTA Setup Procedure 9–4
9.3.4 Add Your Customizations to the New Startup File 9–6
9.3.5 Create the MTA Entry in the Directory 9–6
9.3.6 Start the MTA 9–7
9.3.7 Check MTA$CREATE_CONS_TEMPLATES.NCL 9–7
9.3.8 Updating System Startup Files 9–8
9.3.9 What to Do Next 9–8
9.4 Setting Up an MTA in the Routing Domain for the First Time 9–8
9.4.1 Set Up Access to a DSA 9–8
9.4.1.1 Configure the DSA 9–8
9.4.1.2 Create a Naming Context for the MTA 9–9
9.4.1.3 Create the DUA Defaults File 9–9
9.4.2 Create the Routing Domain and Area Entries for the MTA 9–10
9.4.2.1 Create the MTS Entity for Your Routing Domain 9–10
9.4.2.2 Add Global Domain Identifiers 9–11
9.4.2.3 Create Area Entries 9–12

9.4.3 What to Do Next 9–12
9.5 Adding More MTAs to Your Routing Domain 9–13
9.5.1 Set Up Access to a DSA 9–13
9.5.2 Provide Authorization to Manage the MTA's Routing Information 9–15
9.5.3 Run the MTA Setup Procedure 9–15
9.5.4 Create the MTA Entry in the Directory 9–18
9.5.5 Start the MTA 9–18
9.5.6 Check MTA$CREATE_CONS_TEMPLATES.NCL 9–19
9.5.7 Updating System Startup Files 9–19
9.6 Using the MTA Verification Procedure 9–20
9.7 Keeping the MTA Startup Script Up-to-Date 9–20
9.8 Setting Up an MTA to Contact a Shadow DSA 9–21
9.9 What to Do Next 9–22
9.10 Setting Up Remote Access to the Routing Information in the Directory 9–23

10 Setting Up the Remaining Routing Information

10.1 Routing Information Stored in the Directory 10–1
10.2 Adding Peer MTA Entities at the MTA 10–3
10.3 Adding Agent Entities at the MTA 10–4

11 Setting Up Connections from Agents to the MTA

11.1 Setup for Agents That Use the XAPI Interface 11–1
11.1.1 Agents That Connect Directly to the MTA 11–2
11.1.2 Agents That Use the API Server 11–2
11.2 Setup for Gateways That Use the Shared File Interface 11–4
11.2.1 MAILbus 400 MTA Queues 11–6
11.2.2 Contents of gateway.dat for Shared File 1984 Interface 11–7
11.2.3 Contents of gateway.dat for Shared File 1992 Interface 11–8

12 Example User Agent

12.1 Prerequisites 12–1
12.2 Submitting a Message to the MTA 12–2
12.2.1 Step by Step Instructions for Submitting a Message 12–3
12.2.2 Example 12–3
12.3 Taking Delivery of a Message with MTAmail 12–5
12.3.1 Displaying the Structure of a Delivered Message 12–6
12.4 Errors Reported by MTAmail 12–6
12.4.1 MH_RC_NO_SUCH_CLIENT 12–6

12.4.2 OM_NETWORK_ERROR 12–6
12.4.3 OM_SYSTEM_ERROR 12–6
12.4.4 OM_WRONG_VALUE_LENGTH 12–7
12.4.5 OM_WRONG_VALUE_MAKEUP 12–7

A Postal and Terminal O/R Addresses

A.1 Postal O/R Address A–1
A.2 Terminal O/R Address A–5

B MTS Script for ACME

C MTS Module Entities and Attributes

D Country Codes

Glossary

Index

Examples

11–1 Example gateway.dat for OpenServer 400 NetWare MHS Gateway to X.400 11–8
11–2 Example gateway.dat for ISOCOR X.400 Router for Lotus Notes 11–9
11–3 Example gateway.dat for ISOGATE for cc:MAIL 11–9
12–1 Display of a Message Delivered by MTAmail 12–5

Figures

1–1 How MTAs Use the Directory 1–4
1–2 MAILbus 400 and MAILbus 1–5
1–3 The MTA Module in the EMA Entity Hierarchy 1–7
1–4 The MTS Module in the EMA Entity Hierarchy 1–8
2–1 An X.400 Message Handling System 2–2
2–2 Structure of a Message 2–5
2–3 Structure of an IPM Content 2–6

3–1 Hierarchy for ACME Shoe Corporation 3–2
3–2 Geographical Hierarchy for ACME Shoe Corporation 3–3
3–3 Business Card 3–5
3–4 Recommended Hierarchy for ACME Shoe Corporation 3–6
3–5 ACME Expanding to Invercargill 3–6
4–1 Different Routing Domains 4–2
4–2 Layout of ACME Routing Domain 4–6
4–3 Areas and Area Servers 4–8
4–4 Structure of an MTA Name 4–9
4–5 The ACME Routing Domain Within its X.400 Management Domain 4–14
4–6 ACME Connections Through Intermediate Routing Domains 4–15
4–7 Domain Entries in the Directory 4–16
5–1 How Mnemonic O/R Addresses are Constructed 5–5
5–2 How Numeric O/R Addresses are Constructed 5–6
5–3 Individuals in Your Routing Domain 5–21
5–4 Individuals in Other X.400 Routing Domains 5–24
5–5 Distribution List Expansion 5–28
7–1 MTS Module and Entities 7–2
7–2 MTA Module and Entities 7–4
7–3 Directory Entries for O/R Addresses 7–7
7–4 Directory Entries for Routing Domains 7–11
7–5 Directory Entries for Areas and MTAs 7–15
7–6 Directory Entries for Foreign Addresses 7–17
11–1 MTA Input and Output Queues 11–5
A–1 How Postal O/R Addresses are Constructed A–4
A–2 How Terminal O/R Addresses are Constructed A–8

Tables

4–1 Worksheet for Your Routing Domain 4–20
4–2 Worksheet for Peer MTA Entity 4–22
4–3 Worksheet for Agent Entity 4–26
5–1 O/R Address Attributes 5–8
5–2 O/R Address for Sally Payne 5–16
A–1 Postal O/R Address Attributes and Values A–2

A–2	Terminal O/R Address Attributes and Values	A–5
C–1	MTS Entity Attributes	C–1
C–2	MTA Entity Attributes	C–2
C–3	Area Entity Attributes	C–2
C–4	ORaddress Entity Attributes	C–2
C–5	Domain Entity Attributes	C–5
C–6	Foreign Address Entity Attributes	C–5
D–1	Country Codes	D–1

Preface

Purpose of this Guide

HP MAILbus 400 MTA Planning and Setup provides introductory information about the MAILbus 400 Message Transfer Agent. It also describes the tasks that you need to complete in order to plan and set up a Message Transfer System (MTS) based on the MAILbus 400 Message Transfer Agent.

Prerequisites

If you have not used the MAILbus 400 MTA before, read *MAILbus 400 Getting Started* before reading this guide. *MAILbus 400 Getting Started* leads you through setting up your first MAILbus 400 MTA without the need to completely understand the product.

If you are upgrading from a previous version of the MAILbus 400 MTA, the installation documentation lists the prerequisite software and Part III of *HP MAILbus 400 MTA Planning and Setup* describes how to set up the MAILbus 400 MTA following an upgrade.

You will find it useful to be familiar with the introductory information in *HP Enterprise Directory - Management*, although this information is not critical to understanding or using the MAILbus 400 MTA.

Related Documents

The documentation provided with the MAILbus 400 MTA comprises:

- *MAILbus 400 Getting Started*
- *HP MAILbus 400 MTA Planning and Setup*
- HP MAILbus 400 MTA installation documentation for the appropriate operating system
- *HP MAILbus 400 MTA Tuning and Problem Solving*

- *HP MAILbus 400 MTA Software Product Description* (SPD) for the appropriate operating system
- *HP MAILbus 400 MTA Cover Letter*, where applicable, for the appropriate operating system
- Online Release Notes

In addition to these documents, reference information is available in the *MTA Module Online Help* and *MTS Module Online Help*.

The following documents provide associated information:

- HP Enterprise Directory Service documentation set, and *Directory Module Online Help*
- HP DECnet-Plus documentation
- X.25 documentation
- *CCITT Blue Book Volume VIII Fascicle VIII.7, Data Communication Networks - Message Handling Systems (Recommendations X.400-X.420)*
- *International Standard ISO/IEC 10021, Information Processing Systems - Text Communications - Message-Oriented Text Interchange Systems (MOTIS)*

How to Use this Guide

This guide is divided into three parts:

Use *Part I, Introduction* to familiarize yourself with the X.400 standards and the MAILbus 400 MTA.

Use *Part II, Planning* to plan the O/R addresses and routing information that you need in order to operate an MTS based on MAILbus 400 MTAs. This part also describes the planning required in order for the MTA to use the HP Enterprise Directory Service.

Use *Part III, Setup* to set up the MTS that you have planned. *Part III, Setup* should be read in conjunction with the appropriate installation documentation.

This guide contains a Glossary, which gives brief explanations of terms as used in the MAILbus 400 MTA documentation set.

Readers's Comments

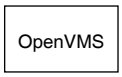
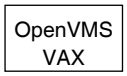
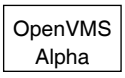
HP welcomes your comments on this manual. Please send comments to either of the following addresses:

Internet	**openvmsdoc@hp.com**
Mail	Hewlett-Packard Company OSSG Documentation Group, ZKO3-4/U08 110 Spit Brook Rd. Nashua, NH 03062-2698

Any software problem that requires an action on the part of HP must be reported to your HP support center, as described in *HP MAILbus 400 MTA Tuning and Problem Solving*. Do not report these problems using the reader's comments addresses.

Conventions

The following conventions are used in this guide:

	Indicates the start of information that is applicable to the Tru64 UNIX operating system.
	Indicates the start of information that is applicable to both the OpenVMS VAX and the OpenVMS Alpha operating systems.
	Indicates the start of information that is applicable only to OpenVMS VAX operating systems.
	Indicates the start of information that is applicable only to OpenVMS Alpha operating systems.
♦	Indicates the end of information that is applicable to a particular operating system.
(Tru64 UNIX)	Where text that refers exclusively to the Tru64 UNIX operating system is minimal, the operating system is indicated in brackets after the text.

(OpenVMS VAX)	Where text that refers exclusively to the OpenVMS VAX operating system is minimal, the operating system is indicated in brackets after the text.
(OpenVMS Alpha)	Where text that refers exclusively to the OpenVMS Alpha operating system is minimal, the operating system is indicated in brackets after the text.
`this typeface`	Indicates commands or responses that you type. Unless otherwise stated, press Return after each command or response.
`this typeface`	Indicates prompts and messages from the computer. Indicates examples of commands.
newterm	Indicates the introduction of a new term.
`variable`	Represents a variable.
UPPERCASE and lowercase	The Tru64 UNIX operating system differentiates between lowercase and uppercase characters. Literal strings that appear in text, examples, syntax descriptions, and function descriptions must be typed exactly as shown.
#	A number sign is the default Tru64 UNIX superuser prompt.
$	A dollar sign is the default OpenVMS prompt.
[]	Brackets are used after questions to enclose the default answer. Press Return to accept the default. Brackets also denote a directory specification.
Ctrl/Z	Indicates that you hold down the Ctrl key while pressing the other key (Z).

Abbreviations

The following abbreviations are used in this guide:

ACSE	Association Control Service Element
API	Application Program Interface
ADMD	Administration Management Domain
ASN.1	Abstract Syntax Notation One
CCITT[1]	International Telegraph and Telephone Consultative Committee
CLNS	Connectionless Network Service
CONS	Connection-oriented Network Service
DIT	Directory Information Tree
DL	Distribution List
DSA	Directory System Agent
DTE	Data Terminal Equipment
DUA	Directory User Agent
EDI	Electronic Data Interchange
EIT	Encoded Information Type
GDI	Global Domain Identifier
IEC	International Electrotechnical Commission
IPM	Interpersonal Message
IPMS	Interpersonal Messaging System
ISO	International Organization for Standardization
MD	Management Domain
MHS	Message Handling System
MPDU	Message Protocol Data Unit
MTA	Message Transfer Agent
MTS	Message Transfer System
NCL	Network Control Language
NFS	Network File Service
NSP	Network Services Protocol
O/R	Originator/Recipient

[1]The CCITT is now the ITU–T (International Telephone Union—Telecommunications). Their published documents still have CCITT identification material, and to avoid confusion this book still uses the term CCITT.

OSI	Open System Interconnection
PC	Personal Computer
PDAU	Physical Delivery Access Unit
PDS	Postal Delivery System
PRMD	Private Management Domain
PTT	Post, Telegraph and Telephone Authority
RDN	Relative Distinguished Name
SMTP	Simple Mail Transfer Protocol
TCP/IP	Transport Control Protocol/Internet Protocol
TP	Transport Protocol

Refer to the HP Enterprise Directory Service documentation for a list of the HP Enterprise Directory Service abbreviations used in this guide.

Part I

Introduction

This part provides an introduction to the MAILbus 400 MTA and X.400.

Part I comprises the following chapters:

- Chapter 1, which explains what the MAILbus 400 Message Transfer Agent (MTA) is and how it can be used.
- Chapter 2, which describes the global messaging system of which the MAILbus 400 MTA is a part.

1

The HP MAILbus 400 Message Transfer Agent

This chapter explains what the MAILbus 400 Message Transfer Agent (MTA) is and how it can be used.

1.1 What is the MAILbus 400 MTA?

The MAILbus 400 MTA provides electronic message transfer services in an open network environment. These message transfer services can be used by applications such as User Agents, Gateways, and Message Stores. A **User Agent** enables users to prepare messages and exchange them with other users. A **Gateway** provides communication between an X.400 and a non-X.400 messaging system. A **Message Store** is an application that stores messages on behalf of its users.

The MAILbus 400 MTA is HP's implementation of an X.400 MTA that conforms to the 1992 messaging standards set up by the International Telegraph and Telephone Consultative Committee (CCITT), the International Organization for Standardization (ISO), and the International Electrotechnical Commission (IEC). These are internationally acknowledged recommendations that were developed to standardize electronic messaging products.

Conformance to these standards makes the MAILbus 400 MTA an "open" product. Open products are based on the concepts of the Open Systems Interconnection (OSI) standards that were developed by ISO to enable different vendors' computer systems to be freely interconnected. An open product can communicate with any other open product that conforms to the same standards.

The MAILbus 400 MTA is the core component of MAILbus 400. The term **MAILbus 400** refers to the MAILbus 400 MTA and the set of open electronic messaging products developed for use with the MAILbus 400 MTA.

1.1.1 The 1992 MHS Standards

In 1984, CCITT introduced the 1984 X.400 Series of Recommendations to support the implementation of a global messaging system. Since then, CCITT, ISO and IEC have worked in close collaboration to develop recommendations and standards to replace the 1984 recommendations.

In 1988, the CCITT issued the 1988 X.400 Series of Recommendations, which are defined in the *CCITT Blue Book Volume VIII Fascicle VIII.7*, and the ISO/IEC issued corresponding X.400 standards defined in the *International Standard ISO/IEC 10021*. For information about how to obtain these documents, see *HP MAILbus 400 MTA Tuning and Problem Solving*. In the MAILbus 400 MTA documentation, these recommendations and standards are referred to as the 1988 Message Handling System Standards, or **1988 MHS Standards**.

Since 1988, the CCITT and ISO/IEC have continually revised their work. In 1992, the CCITT have consolidated their revisions with the 1988 X.400 Series of Recommendations to form the 1992 editions of these recommendations. For the purpose of the MTA documentation, the 1988 CCITT X.400 Recommendations, International Standard ISO/IEC 10021, *and* the revisions to these recommendations and standards are collectively called the **1992 MHS Standards**. Any reference in the MTA documentation to the 1992 MHS Standards includes the 1988 MHS Standards, unless indicated otherwise. *HP MAILbus 400 MTA Tuning and Problem Solving* contains a detailed list of the revision documents that, combined with the 1988 MHS Standards, make up the 1992 MHS Standards.

The 1984 X.400 recommendations are referred to as **1984 MHS Standards** in the MTA documentation. Note also that the term "X.400" is used in the MTA documentation to refer to both the 1984 and the 1992 MHS Standards. Any exceptions are specifically defined.

The MAILbus 400 MTA conforms to the 1992 MHS Standards. In places where the CCITT recommendations and the ISO standards differ, the MAILbus 400 MTA product implements the ISO standard.

The MAILbus 400 MTA is capable of interworking with messaging systems conforming to the 1984, 1988 or 1992 MHS Standards. The MAILbus 400 MTA also conforms as closely as possible to the relevant profiles being developed in Europe, the USA, and South East Asia. See *HP MAILbus 400 MTA Tuning and Problem Solving* for further information about the MAILbus 400 MTA's conformance to the 1992 MHS Standards and related profiles, and for information on how it interworks with 1984 messaging systems.

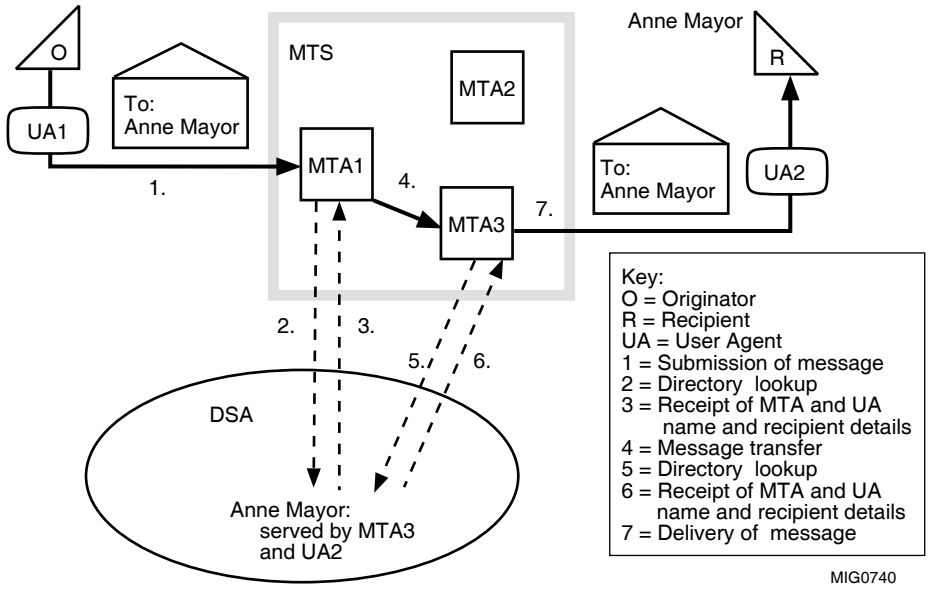
1.2 The MAILbus 400 MTA and the HP Enterprise Directory Service

For the MAILbus 400 MTA to operate it requires a **directory**. The directory is used by a number of MAILbus 400 MTAs allowing them to share the information they need in order that users and applications can exchange messages. The user or application sending the message is called the **originator**. The user or application receiving the message is called the **recipient**.

The directory used by the MAILbus 400 MTA is the HP Enterprise Directory Service. This is a directory service implementation based on the 1993 edition of the CCITT X.500 Series of Recommendations and on International Standard ISO/IEC 9594. It comprises a directory system agent (DSA) and a directory user agent (DUA). The **DSA** is the part of the HP Enterprise Directory Service that is responsible for storing information and providing access to it. The **DUA** is the interface that the MAILbus 400 MTAs use to access the information held in the directory. A DUA is integrated with each MAILbus 400 MTA.

MTAs use the information stored in the directory to route messages to their intended recipients. **Routing** is the process by which MTAs use an O/R address specified in the message envelope to look in the directory for routing information about how to reach this address. An **O/R address** is a hierarchical set of attributes that uniquely identifies a place or system where a user receives electronic mail. For more information about O/R addresses refer to Chapter 5. Figure 1–1 illustrates a simplified directory lookup.

Figure 1–1 How MTAs Use the Directory



Part II of *HP MAILbus 400 MTA Planning and Setup* describes how to plan your directory information. Note that throughout the MAILbus 400 MTA documentation set, the HP Enterprise Directory Service is referred to as the directory.

1.3 The MAILbus 400 MTA and MAILbus

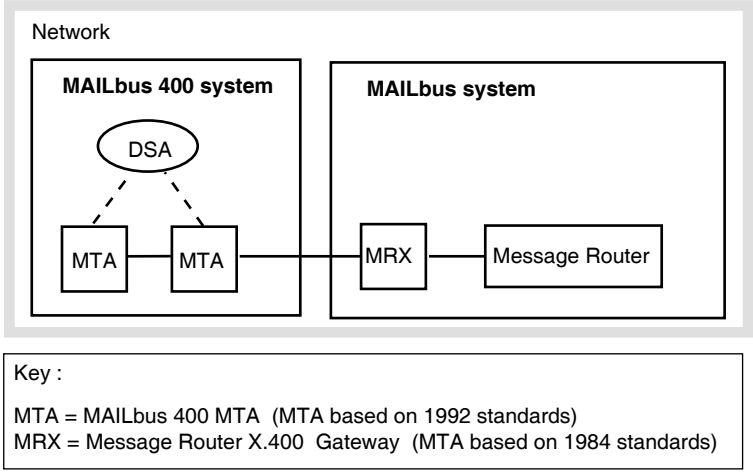
MAILbus is the term used to refer to the set of HP messaging products that provide the backbone of an electronic messaging system. Examples of MAILbus products are Message Router and its Gateways, such as the Message Router VMSmail Gateway or the Message Router X.400 Gateway (MRX). MAILbus products are based on international standards that have since been superseded. For a description of MAILbus and MAILbus components see *Introduction to MAILbus*[1].

You can connect the MAILbus 400 MTA to an existing MAILbus system through either the Message Router X.400 Gateway (MRX) or the MAILbus 400 Message Router Gateway.

[1] This guide is part of the VAX Message Router kit; its order number is AA-Q1RK*-TE.

MRX functions as a Message Transfer Agent that conforms to the CCITT 1984 X.400 Recommendations. Therefore, a connection between the MAILbus 400 MTA and MRX is a connection between two X.400 MTAs, not a connection between an X.400 MTA and a Gateway. Figure 1–2 illustrates a connection between the MAILbus 400 MTA and MRX.

Figure 1–2 MAILbus 400 and MAILbus



In contrast, a connection between the MAILbus 400 MTA and the MAILbus 400 Message Router Gateway is an MTA to Gateway connection. For more information about MAILbus 400 Message Router Gateway see the MAILbus 400 Message Router Gateway documentation set.

Chapter 4 contains more information about how the MAILbus 400 MTA can be connected to other X.400 and non-X.400 messaging systems.

1.4 Operating Over DECnet-Plus and TCP/IP Networks

The MAILbus 400 MTA can operate over both HP DECnet-Plus and TCP/IP networks. *HP MAILbus 400 MTA Tuning and Problem Solving* describes in detail how the MAILbus 400 MTA operates over these two types of network. The following sections provide a brief overview of the networks.

1.4.1 The MAILbus 400 MTA and HP DECnet-Plus

The MAILbus 400 MTA is a HP DECnet-Plus application.

HP DECnet-Plus is the HP hardware and software that implements the DIGITAL Network Architecture (DNA) Phase V and supports Open Systems Interconnection (OSI) protocols. HP DECnet-Plus was formerly DECnet/OSI.

OSI defines a seven-layer reference model for a network architecture. Starting from the top, the layers are: Application, Presentation, Session, Transport, Network, Data link, and Physical. Each layer provides services to the layer above and uses the services provided by the layer below. For more information about HP DECnet-Plus and the OSI reference model see DECnet-Plus documentation set.

As an application in the Application layer of HP DECnet-Plus, the MAILbus 400 MTA uses relevant services from all lower layers of HP DECnet-Plus. For example, the MAILbus 400 MTA can be set up to use either of the network services provided by the Network layer for MTA-to-MTA connections:

- Connection-oriented network service (CONS)
- Connectionlessnetwork service (CLNS)

At the Transport layer, the MAILbus 400 MTA's use of OSI transport protocols for MTA-to-MTA connections is restricted to the following:

- OSI Transport Protocol Class 0 (TP0)
- OSI Transport Protocol Class 2 (TP2)
- OSI Transport Protocol Class 4 (TP4)

1.4.2 The MAILbus 400 MTA and TCP/IP

The MAILbus 400 MTA can also connect to other MTAs over TCP/IP networks. It does this by using the RFC 1006 protocol, which defines how to emulate the OSI Transport Service operating OSI Transport Protocol Class 0 (TP0) over a TCP/IP network service.

1.5 Conformance to the Enterprise Management Architecture

As an application layered on DECnet-Plus the MTA conforms to the Enterprise Management Architecture (EMA), which is designed to integrate management of an entire network based on HP DECnet-Plus. Management of the directory is also based on EMA.

Conformance to EMA gives you a number of advantages when managing the MTA and the information in the directory:

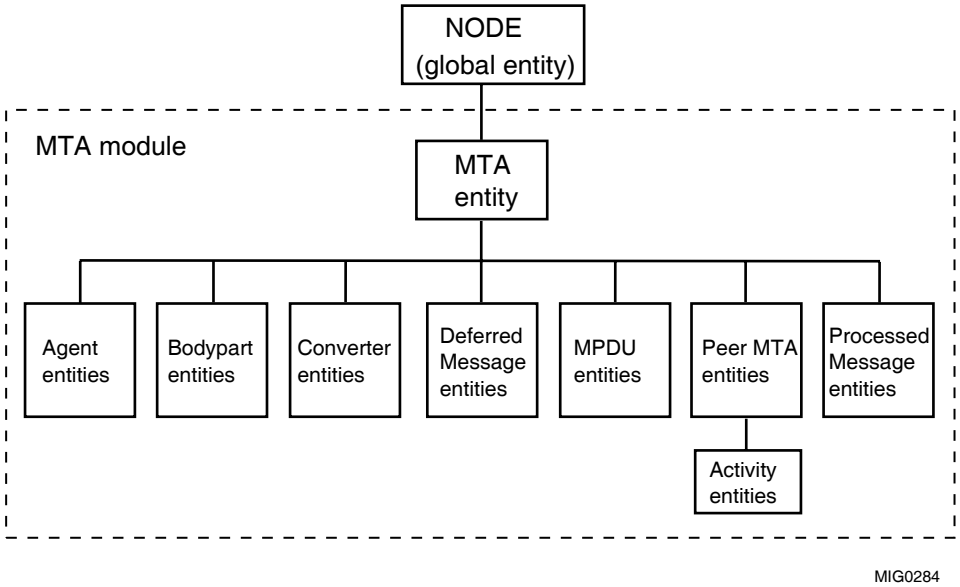
- The MTA and its routing information can be managed remotely.
- Management responsibility can be distributed throughout the network.
- Management of the MTA and the directory information is consistent with management of other HP DECnet-Plus applications.

Within EMA, the MTA and directory information used by the MTA are represented as management modules:

- **MTA module**

  The MTA is a module directly below the global entity NODE. The MTA module consists of the MTA entity and its subordinate entities, as shown in Figure 1–3. These entities enable you to manage the operation of a particular MTA. More detailed information about the entities of the MTA module is available in the *MTA Module Online Help* and Part I of *HP MAILbus 400 MTA Tuning and Problem Solving*.

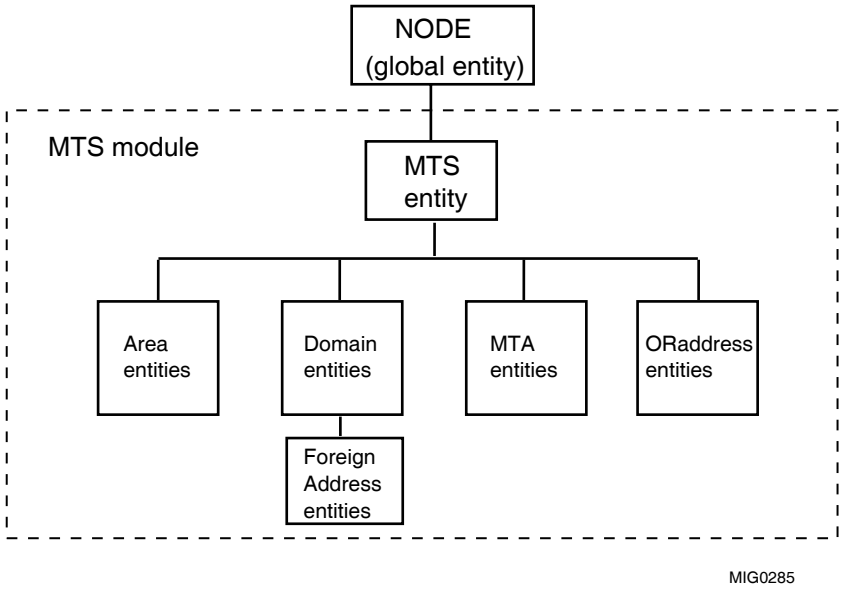
Figure 1–3 The MTA Module in the EMA Entity Hierarchy



- **MTS module**

  The information that is shared by all MTAs within a routing domain is held in the directory. This information is represented by the entities of the MTS module. The MTS module is a subentity of the global entity NODE and consists of the MTS entity and its subentities, as shown in Figure 1–4. You use these entities to enter and manage the information in the directory.

Figure 1–4 The MTS Module in the EMA Entity Hierarchy



The entities of the MTS module are different from other EMA entities in that they do not hold information themselves. They are the interface through which you enter information in the directory. Because the entities of the MTS module do not hold information, they do not have counter and status attributes, and do not issue events. However, events are issued by the MTA entity of the MTA module if it finds inconsistencies in the information that the directory holds.

Even though the MTS module is a module below the global entity NODE, the information represented by its entities is not specific to that node. The information is shared by the MTAs in the entire routing domain. For more information about the MTS module see the *MTS Module Online Help* and Part II of *HP MAILbus 400 MTA Planning and Setup*.

You can use any EMA compliant director to manage the entities of the MTA and MTS modules, for example, HP's Network Control Language (NCL). Other management tools that can be used with the MTA have graphical user interfaces. These are:

- The TeMIP™ product family, which can be used to manage the MTA module information

NCL is a command line interface and is installed on every DECnet-Plus node. You can enter NCL commands interactively from a terminal or execute them as part of a script. For a description of the NCL command line interface, refer to the *NCL Online Help* and the DECnet-Plus network management documentation. You can access the *MTS Module Online Help* and *MTA Module Online Help* as topics in the *NCL Online Help*.

1.6 What to Do Next?

You should now read Chapter 2 for information about the global messaging system of which the MAILbus 400 MTA is a part.

2

An X.400 Message Handling System and its Components

In the context of the 1992 MHS Standards, a **message handling system (MHS)** is a messaging system that enables a user (or an application) to exchange messages with one or more other users (or applications). An application is defined as hardware and software that performs a specific task. The user or application sending a message is called the originator, and the user or application receiving a message is called the recipient.

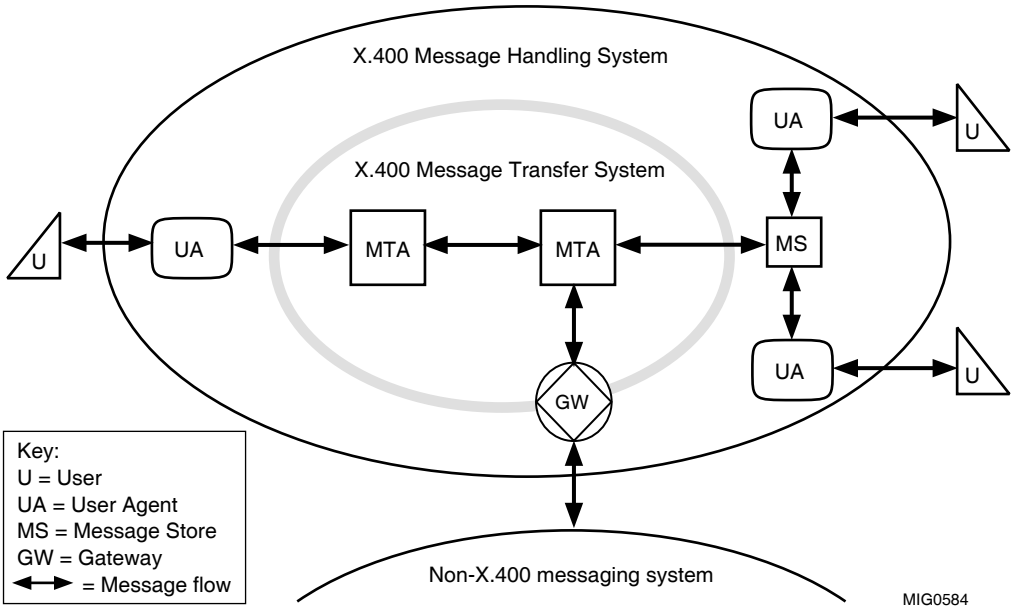
An X.400 MHS consists of the following components:

- Message Transfer Agents (MTAs)(see Section 2.1)
- User Agents (see Section 2.2)
- Access Units, which are more commonly known as Gateways (see Section 2.3)
- Message Stores (see Section 2.4)

The MAILbus 400 MTA documentation refers to User Agents, Gateways, and Message Stores collectively as **Agents**.

Figure 2–1 illustrates an X.400 MHS including MTAs, User Agents, a Message Store, and a Gateway.

Figure 2–1 An X.400 Message Handling System



2.1 The Role of the Message Transfer Agent

Message Transfer Agents are the core of an X.400 MHS. MTAs operate together to transfer messages so that they can be delivered to their destinations. A group of connected MTAs is called a **message transfer system (MTS)**.

An MTA transfers a message to another MTA by first establishing an **association** with that MTA. MTAs transfer messages in a **store-and-forward** manner. For example, if one MTA attempts to transfer a message to another MTA that is temporarily unavailable, the first MTA stores the message until transfer is possible. An MTA can communicate with any number of User Agents, Access Units, and other X.400-compliant MTAs.

The MAILbus 400 MTA is HP's implementation of such an MTA.

2.2 The Role of the User Agent

An application that enables a user to interact with an MTS is called a User Agent. It enables the user to prepare messages to send to other users and displays messages received from other users. The process by which a User Agent passes a message to an MTA is called **submission**, and the process by which an MTA passes a message to a User Agent is called **delivery**.

An example of X.400 User Agents provided by HP is HP Office Server.

2.3 The Role of the Access Unit

An application that links an X.400 MHS to another messaging system, for example a non-X.400 paper mail delivery system or a telex network, is called an **Access Unit**, or more commonly, a Gateway.

A Gateway in the X.400 MHS enables a user (or application) to exchange messages with users (or applications) of other, typically non-X.400 messaging systems. A Gateway translates messages from the encoding method used in an X.400 MHS to the encoding method used in the other messaging system. Likewise, it translates messages from the encoding method used in the other messaging system to X.400 format. The process by which an MTA passes a message to a Gateway is called **export**, and the process by which an MTA receives a message from a Gateway is called **import**.

An example of a Gateway provided by HP is the HP MAILbus 400 SMTP Gateway, which provides access from an X.400 MHS to the Internet.

2.4 The Role of the Message Store

A Message Store is an application that provides message storage and submission services for the users of an X.400 MHS.

A Message Store takes delivery of messages from an MTA, and stores them on behalf of the Message Store users. The users can connect to the Message Store with their User Agents at any time to retrieve their messages. A Message Store also submits messages to the MTA on behalf of its users.

The MAILbus 400 Message Store is HP's implementation of an X.400 Message Store.

2.5 Messages, Reports and Probes in an X.400 MHS

An X.400 MHS is designed to transfer messages between its users and/or applications. In addition to messages, MTAs in an X.400 MHS also transfer:

- Information about the status of messages, in the form of reports (see Section 2.5.2)
- Condensed representations of messages, in the form of probes (see Section 2.5.3)

In the MAILbus 400 MTA documentation, the term "message" is used in a general way and refers to messages, reports and probes. Any exceptions are specifically defined. Section 2.5.1, Section 2.5.2 and Section 2.5.3 describe messages, reports, and probes respectively.

2.5.1 What is a Message?

Messages are those items of communication that are exchanged between users and/or applications in a messaging system.

2.5.1.1 Structure of a Message

A message in an X.400 MHS consists of an envelope and a content, as shown in Figure 2–2 and explained in the following list:

- **Envelope**

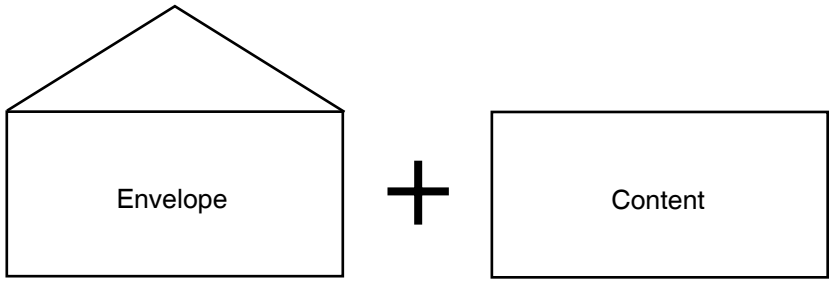
  The envelope of a message provides the information that the X.400 MTAs require to transfer the message and deliver it to the intended recipients.

  The envelope contains information such as the mail addresses of the originator and the recipients, and the priority with which the MTS is to process a message. Information contained in the envelope is regulated by the Message Transfer Protocol P1, which is defined in CCITT Recommendation X.419 and International Standard ISO/IEC 10021-6.

- **Content**

  The content of a message contains the piece of information the originator wants to send to the recipient. Generally, the information in the content of a message is not accessed by MTAs when transferring the message. MAILbus 400 MTAs only access a message content if the message is an interpersonal message and its content structure or character set must be converted before it can be delivered to the recipient's Agent. See Section 2.5.1.2 for information about interpersonal messages.

Figure 2–2 Structure of a Message



2.5.1.2 Interpersonal Messages

Messages created by mail User Agent applications and exchanged between users in an X.400 MHS are referred to as **interpersonal messages (IPMs)**. These messages are of the interpersonal messaging system (IPMS) content type.

The **content type** of a message is indicated in the message envelope and defines the syntax and the semantics of the data in the content. Another example of a content type is the electronic data interchange (EDI) content type. See the DEC/EDI documentation set for more information about EDI messages.

An IPM has a defined structure, which is regulated by the Interpersonal Messaging Protocol P2 as defined in CCITT Recommendation X.420 and International Standard ISO/IEC 10021-7. The content of an IPM consists of a heading and a body, as shown in Figure 2–3 and described in the following list:

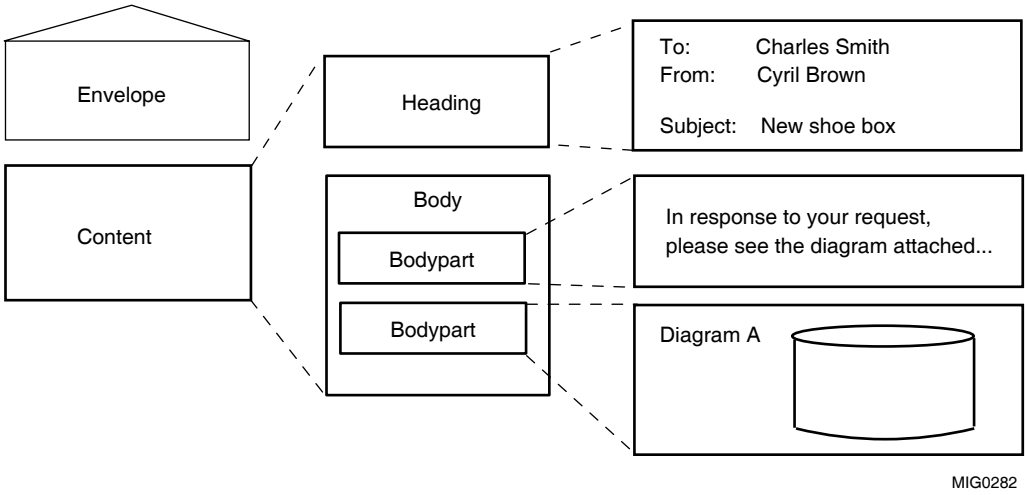
- **Heading**

  The heading consists of a set of heading fields. These fields contain information about the IPM, such as the name of the originator and the recipient, and the subject of the message.

- **Body**

  The body consists of a series of **bodyparts**. An IPM bodypart contains a piece of information to be transferred in the IPM. MTAs can transfer a wide range of bodypart types within a single IPM, for example, text, facsimile, videotext or a forwarded message.

Figure 2–3 Structure of an IPM Content



The MAILbus 400 MTA transfers messages of all content types, and converts bodyparts of an IPM if conversion is required before delivery to a User Agent or Message Store, or before export to a Gateway. The MAILbus 400 MTA only converts IPM bodyparts. It does not convert messages with any other content type.

2.5.2 What is a Report?

A **report** is generated by an MTA in the MTS. It contains information about the status of a message or probe, and is sent to the originator of the message or probe. See Section 2.5.3 for a definition of probes.

A report is sent, for example, when a message is incorrectly addressed and the MTA cannot identify an intended recipient. Without sufficient address information an MTS is unable to transfer or deliver the message. When a message cannot be delivered, the MTS sends a **non-delivery report** to the message originator. Likewise, the MTS can also return a **delivery report** to notify the originator when a message or probe has reached its destination.

A report consists of an envelope and, if required by the originator, the content of the subject message.

2.5.3 What is a Probe?

A **probe** is a condensed representation of a message. An originator can send a probe to find out whether a message can be delivered to its intended recipients.

A probe consists only of an envelope, which defines all the characteristics of the message it represents. When a probe is submitted to an MTA, it is transferred from the originator's MTA to the MTAs serving the recipients listed on the message. The probe is not actually delivered. The probe causes the recipients' MTAs to generate the same reports (that is, delivery reports or non-delivery reports) as they would generate for the message that the probe represents.

If the MTS fails to transfer the probe successfully, it is probably unable to deliver the associated message. The successful transfer of the probe to all intended destinations provides a strong indication that the associated message can reach its recipients, but does not guarantee it.

2.6 What to do Next

Continue and read Part II of *HP MAILbus 400 MTA Planning and Setup*, which describes the planning tasks that you need to complete before setting up your message transfer system.

Part II
Planning

This part describes the planning tasks that you need to complete in order to have an operational Message Transfer System (MTS) based on the MAILbus 400 MTA.

Part II comprises the following chapters:

- Chapter 3, which describes how to plan a naming scheme for MTAs and O/R addresses in your organization.
- Chapter 4, which describes how to plan names for MTAs, Agents, and routing domains.
- Chapter 5, which describes how to plan O/R addresses for individuals in your routing domain. This chapter also explains why you need to include the O/R addresses for individuals in other routing domains, and how to assign routing information to all O/R addresses.
- Chapter 6, which describes what needs to be planned in order for the MAILbus 400 MTA to use the HP Enterprise Directory Service.
- Chapter 7, which provides an example of the routing information planned for the fictitious ACME Shoe Corporation and how this information is entered in the directory.

Some of the tables in Part II indicate the character sets that you can use to specify the values of attributes of the management entities. For information about the characters contained in each character set, see *HP MAILbus 400 MTA Tuning and Problem Solving*. The *MTA Module Online Help* and *MTS Module Online Help* also give information about the character sets.

3
Planning a Naming Scheme

This chapter describes how to design a naming scheme for assigning O/R addresses to individuals in your organization, or in a part of your organization. The naming scheme that you choose can also be used to name MTAs in your organization.

To plan an MTS for your organization, it helps to have a good understanding of the organization's size and structure. For example, the organization might be using a number of different electronic mail systems. You can plan an MTS made up of MAILbus 400 MTAs to link together groups within your organization that use a variety of electronic mail systems.

3.1 Defining a Hierarchy for Your Organization

To simplify the process of planning O/R address attribute values and MTA names, it is helpful to have an understanding of your organization's hierarchy. This is because O/R address attribute values are ordered hierarchically and, if you have a large organization, you might want to use this hierarchy in your naming scheme. If you have a small organization, with no formal hierarchy, you can choose not to plan a hierarchical naming scheme.

An organization can usually be divided into smaller units, such as departments, or sites. These divisions can be either organizational or geographical. Organizational divisions represent the ways in which your organization, or its subdivisions, can be divided into groups according to the different roles or functions that each group has. Geographical divisions represent the ways in which your organization, or its subdivisions, can be divided into groups according to location, for example, countries, regions, states, counties, towns, and buildings.

Figure 3–1 Hierarchy for ACME Shoe Corporation

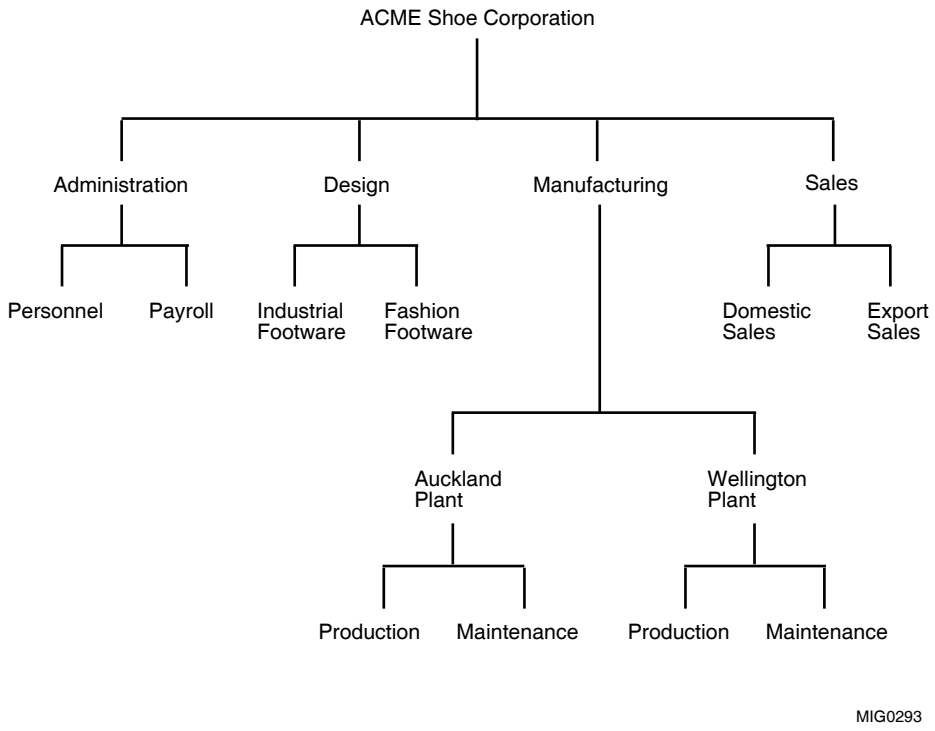
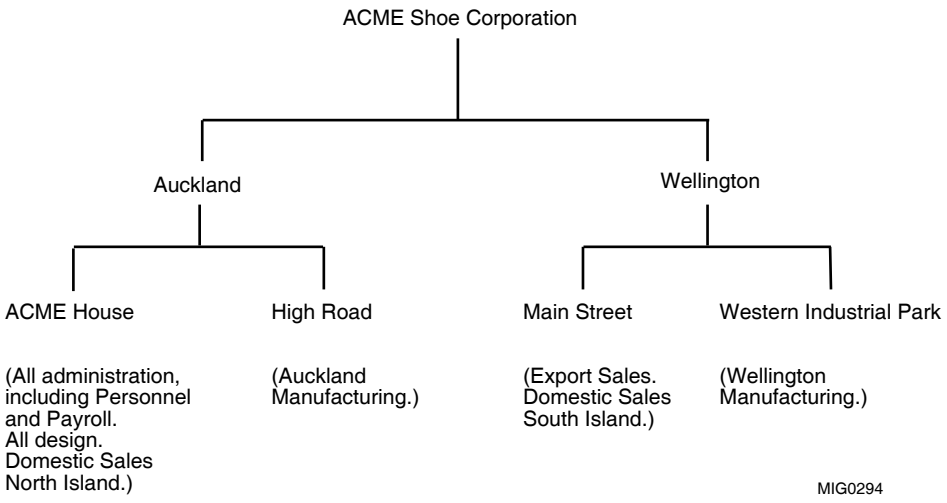


Figure 3–1 shows the hierarchy of a fictitious company, the ACME Shoe Corporation, a shoe manufacturer in New Zealand. This hierarchy is based on both the geographical and the organizational structure of the organization. Examples of organizational divisions are Administration and Manufacturing. Examples of geographical divisions are Auckland Plant and Wellington Plant.

Figure 3–2 shows the geographical hierarchy of the same organization. The diagram also shows which departments are situated at each location.

You might find it useful to create similar diagrams to represent your organization's hierarchy and to show the full depth of the hierarchy. Later on in your planning you may decide to disregard some of the levels, but it is helpful to start with a complete picture.

Figure 3–2 Geographical Hierarchy for ACME Shoe Corporation



Section 3.2 describes what you should consider when planning a naming scheme for use as the basis of your O/R addresses and MTA names.

3.2 Choosing a Naming Scheme

Invest some effort in working out a naming scheme that is appropriate to your organization before you allocate O/R addresses and MTA names. In this way, you avoid the need to make changes after you have implemented a scheme.

You do not have to represent the full depth of your hierarchy in your naming scheme. Some organizations may not have a hierarchy because the number of individuals in the organization is not large enough.

Consider the following when choosing a naming scheme:

- Familiar naming schemes

  Most organizations have a method of uniquely identifying the departments, locations, and people within them. Study any current naming schemes that your organization uses and decide whether you want to incorporate them into your naming scheme.

  As an example, there could be a paper mail system in use in your organization that you could use as the basis for a naming scheme.

- Geographically based naming schemes

  A geographical naming scheme is likely to be more stable than an organizational naming scheme, because in most organizations it is more common for people to change groups rather than locations.

  For example, the ACME Shoe Corporation, whose geographical hierarchy is shown in Figure 3–2, could design a naming scheme that takes into account both levels of its geographical hierarchy, that is, towns and buildings. However, if there is a possibility that ACME will move its operations to different buildings, it would be best for them not to represent buildings in their naming scheme.

- Levels in the naming hierarchy

  O/R addresses and MTA names are hierarchical and are assigned on the basis of the naming scheme that you choose. If you choose a naming scheme based on a deep naming hierarchy, the names are longer and difficult to remember, and therefore less usable.

  You might find that some Gateways need an O/R address to represent their user communities, for example, the MAILbus 400 Message Router Gateway and the HP MAILbus 400 SMTP Gateway. You will need to take this into account when designing a naming scheme for your organization. Refer to the documentation associated with these Gateway products to find out what their requirements are.

- Reducing the possibility of ambiguities in the naming scheme

  Ambiguities occur when there are two or more users with the same name immediately under the same point in the hierarchy. The fewer levels of hierarchy that you have, the more likely you are to have naming ambiguities.

  Ambiguities are more likely to occur when assigning individual O/R addresses than when assigning MTA names. Section 5.4.1 provides some guidelines about resolving naming ambiguities for O/R addresses.

- Confidentiality

  Use a naming scheme that you are happy to expose outside your organization, for example, on business cards (see Figure 3–3). This is particularly important if you intend to base your naming scheme on organizational structure. By exposing your organization's hierarchy in your O/R addresses, you are making it easier for others to understand the internal structure of your organization.

Figure 3–3 Business Card

- Name length

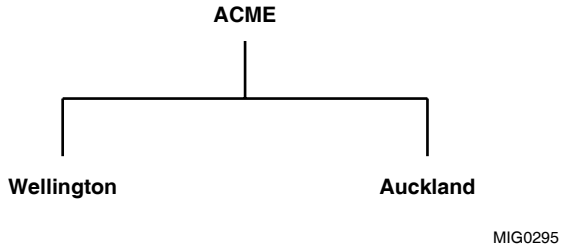
  Keep names short, as they are easier to remember and less prone to spelling errors.

Some of these recommendations conflict. Choosing a suitable naming scheme may involve compromising one consideration in favor of another.

Base MTA names on a geographical hierarchy, and base O/R addresses on either a geographical or organizational hierarchy. In our experience we have found that using a familiar naming scheme, based on a geographical organization, with few levels of hierarchy, is suitable for both O/R addresses and MTA names.

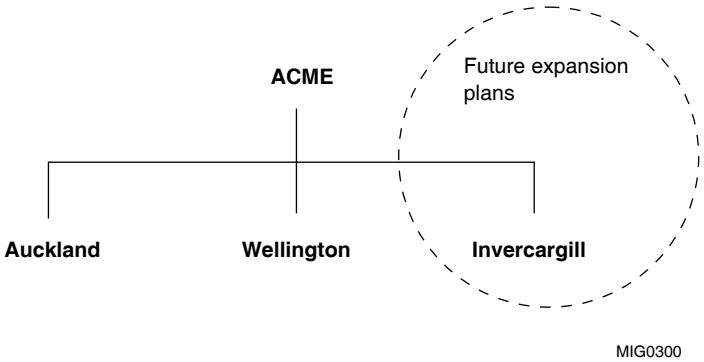
Figure 3–1 and Figure 3–2 show two example hierarchies for the ACME Shoe Corporation. Figure 3–4 shows the recommended hierarchical structure for the ACME Shoe Corporation to use as a basis for assigning O/R addresses and MTA names. This hierarchical structure was chosen based on the fact that ACME is located at two different geographical locations, and might be divided into areas, and also because the single level of hierarchy based on location is sufficient to represent O/R addresses.

Figure 3–4 Recommended Hierarchy for ACME Shoe Corporation



You might only be planning a naming scheme for part of an organization, in which case there are fewer constraints on your naming scheme. However, still consider a naming scheme that you can expand in the future. On the other hand, you might be designing a naming scheme that will be used throughout the organization and that will need to take account of several discrete parts of the organization. Figure 3–5 shows how ACME could later expand to another town in New Zealand, Invercargill, without the need to change this naming scheme.

Figure 3–5 ACME Expanding to Invercargill



4

Naming Routing Domains, MTAs and Agents

This chapter describes how to plan names for MTAs, Agents, and routing domains. In the case of MAILbus 400 MTAs, most of this information is held in the directory. However, some information is held locally at individual MTAs.

MTA names have a hierarchical structure that includes area names. You need to understand this structure before you allocate names to the MTAs in your routing domain.

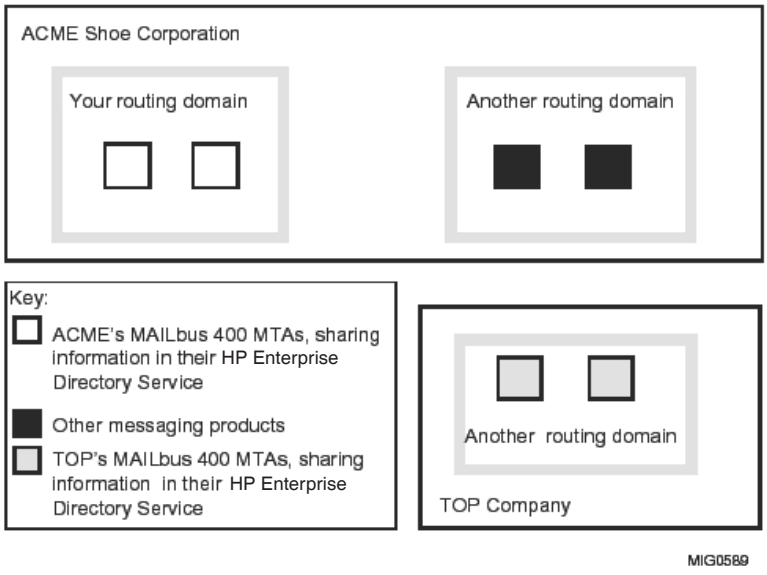
At the end of this chapter is a series of worksheets, in the form of tables, that you can use to fill in name details as you plan. You can use these worksheets when setting up your routing domain.

4.1 Routing Domains

In the context of the MAILbus 400 MTA, a group of messaging products that use the same information in order to transfer messages is called a **routing domain**. This means that the MAILbus 400 MTAs that share routing information in the directory constitute one routing domain. In the MTA documentation, your MAILbus 400 MTAs that share routing information in the directory are referred to as your routing domain.

In addition to your routing domain, which consists of a group of MAILbus 400 MTAs, you might have other routing domains in your messaging network. The following list describes examples of other routing domains. The examples are based on the ACME Shoe Corporation in New Zealand. Figure 4–1 illustrates the difference between your routing domain and other routing domains.

Figure 4–1 Different Routing Domains



Other routing domains can be:

- Routing domains consisting of other messaging products

  These could be routing domains within your X.400 management domain or within another X.400 management domain. You can view other X.400 management domains as being other organizations. Section 5.1.2 explains X.400 management domains in detail.

  For example, your organization, ACME Shoe Corporation, uses MAILbus 400 MTAs, as well as other messaging products. Therefore, different groups of people in ACME Shoe Corporation use different messaging products. The MAILbus 400 MTAs in ACME share routing information contained in the directory, while the other messaging products use their own information. The other messaging products are therefore not part of your routing domain. The user community of ACME Shoe Corporation is divided into more than one routing domain.

- Routing domains consisting of MAILbus 400 MTAs that access different routing information

  For example, ACME Shoe Corporation has close business ties with another organization, called the TOP Company. TOP Company is a PRMD which also uses MAILbus 400 MTAs. However, these MTAs use their own routing information stored in their own DSA and do not share the routing information stored in the DSA used by ACME's MAILbus 400 MTAs. They are therefore in another routing domain.

  Another example is where ACME Shoe Corporation uses only MAILbus 400 MTAs, but different groups of these MTAs are set up to access different routing information in the directory.

Summary

Your routing domain consists of all the MAILbus 400 MTAs that share the same routing information in the directory. Any MTA that does not share this information is considered to be in another routing domain.

You can subdivide your routing domain into areas in order to optimize message transfer within your routing domain. This is described in Section 4.2.2. You can also connect your routing domain to other routing domains. This is described in Section 4.5.

4.1.1 Naming Your Routing Domain

If you have previously set up your first MTA using *MAILbus 400 Getting Started*, you will already have created entries for your routing domain, in which case you can continue at Section 4.2.

You need to plan a name and password for your routing domain. The name is part of the distinguished name for the routing domain entry, which is the routing domain's entry in the directory. The password protects your routing information from unauthorized users.

Subordinate to the entry describing your routing domain is all the routing information that relates to the routing domain. The routing domain entry can itself be subordinate to the root of the HP Enterprise Directory Service Directory Information Tree (DIT) or any of the following HP Enterprise Directory Service object classes:

- country
- locality

- organization
- organizationalUnit

Refer to the HP Enterprise Directory Service documentation for definitions of these object classes. Note that the examples provided in this book assume that the routing domain entry is subordinate to the root of the DIT.

Contact the person responsible for managing the HP Enterprise Directory Service to ensure that the name that you choose for the routing domain is unique within the directory. If there is no-one responsible for managing the directory, choose a name according to the guidelines given in *HP Enterprise Directory - Management*.

The relative distinguished name (RDN) for the routing domain entry has an attribute type of **MTS** and the attribute value that you are allocated, or that you choose. As an example, the distinguished name for the ACME Shoe Corporation's routing domain is "/MTS=ACME".

The distinguished name for the routing domain is used as the identifier of the MTS entity, an entity of the MTS module. The MTS module is the management module that represents the routing information for your routing domain in the directory.

The routing domain name has no significance other than uniquely identifying the routing domain in the directory, and thus identifying the location of the MTA routing information within the directory.

4.1.2 Global Domain Identifier

In addition to a name for your routing domain, you need to identify the **global domain identifiers** (GDIs) for your routing domain. GDIs are allocated to every X.400 management domain and identify your routing domain within the global MHS. A GDI is an O/R address comprising: the Country Name attribute, the Administration Domain Name attribute and, optionally, the Private Management Domain Name attribute. Typically, the Administration Domain Name will be the name of your service provider and the Private Management Domain Name will be the name of your organization.

If you are managing a routing domain that connects to more than one X.400 management domain it will have more than one GDI. A routing domain that has more than one GDI is referred to as being **multi-homed**. The MTAs within a routing domain need to know the GDIs that are applicable to the routing domain when adding trace information to a message.

The GDIs that are allocated to your routing domain are part of the O/R addresses for the individuals in your routing domain. Chapter 5 describes O/R address attributes, including the GDI attributes in more detail.

You must make sure that you list all the GDIs that are applicable to your routing domain as attributes of the MTS entry in the directory.

4.2 Understanding the Layout of Your Routing Domain

You also need to understand how MTAs in your routing domain communicate with other parts of the MHS. The easiest way to do this is to draw a plan of your routing domain that includes MTAs and User Agents. At a later stage you can add to the plan other Agents and routing domains that your MTAs connect to.

Figure 4–2 shows the layout of the ACME routing domain and includes both registered and unregistered User Agents. Section 4.4 describes how to plan names for the Agents in your routing domain.

Naming MTAs is described in Section 4.3. Before you name individual MTAs, you need to plan whether you want to use MTA sets and whether you want to divide your routing domain into areas. Section 4.2.1 describes MTA sets. Section 4.2.2 describes why you might choose to divide your routing domain into areas, and what impact this has on naming your MTAs.

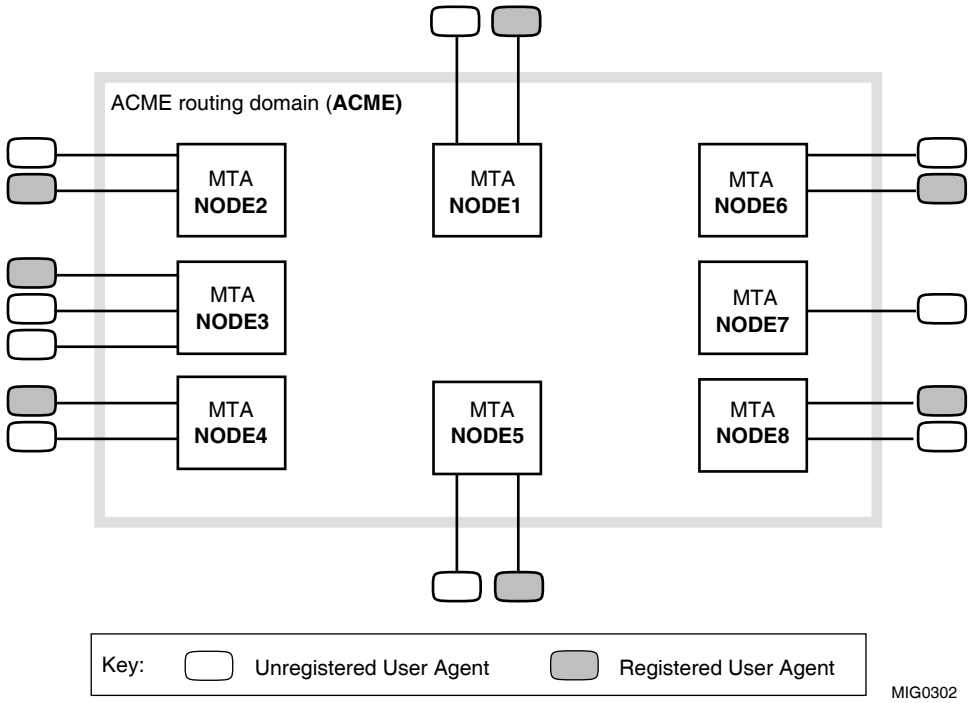
Details of the names and the characteristics of the MTAs in your routing domain are held in the directory and are available to all other MTAs within the routing domain for use when routing messages.

4.2.1 MTA Sets

In most situations where you decide to use an MTA, you can choose to use an MTA set. An MTA set is a number of MTAs that are set up so that they are equivalent in terms of the service they are providing.

Use MTA sets where it is important that you have a high availability of service in your network, for example, as an area server (see Section 4.2.2) or as an MTA that connects to another routing domain (see Section 4.5). In this way, if one MTA in the set becomes unavailable, the remaining MTAs can still accept messages for transfer. (An MTA that connects to an MTA in another routing domain is known as a **boundary MTA**.)

Figure 4–2 Layout of ACME Routing Domain



In most cases you will want all the MTAs in a set to be set up identically to simplify the management of the MTA set. This reduces the possibility of inconsistencies of operation for each of the MTAs. However, in some circumstances, you might want to set up the characteristics of the MTAs in the set to be slightly different, as the systems on which the MTAs operate are different. For example, you might choose to vary the number of connections each MTA in the set can accept.

If you use an MTA set to serve an Agent, you must ensure that each MTA in the set can contact the Agent. The Agent must also be capable of communicating with all the MTAs in the set in order to receive messages.

If you plan to use an MTA set, you require:

A name for the MTA set.
The list of MTA names that make up the MTA set.

This information is stored in the directory.

Section 4.3 describes how to allocate names to the MTAs and MTA sets in your routing domain according to the naming scheme chosen in Chapter 3. However, before you allocate names to your MTAs, consider whether or not you want to divide your network into areas.

4.2.2 Dividing Your Routing Domain into Areas

An **area** consists of a group of MTAs that are part of a particular geographical location. Splitting your routing domain into areas can optimize message transfer across large and geographically dispersed areas. An area could be a site, a country or another geographical area.

You might choose to divide your routing domain into areas if it has communications links that are expensive to use, such as international or satellite links, or links that are congested.

If you want to implement routing based on areas, identify one of the MTAs in an area to be the **area server**. This MTA needs to be on a node that has the capacity to handle the incoming messages, potentially for the whole area. This is because, when an MTA in your routing domain transfers a message to multiple recipients, it creates separate instances of this message, as many instances as are required to deliver the message to the recipients. This process is called **message splitting**, and the separate instances of a message are called **message protocol data units** (MPDUs). The MPDUs are then transferred directly to the recipients' MTAs.

Using areas minimizes the use of expensive communications links between areas for messages sent from one area and addressed to a number of recipients at different MTAs in another area. In this way, message transfer to recipients within an area is optimized such that one copy of the message is sent to the area server, and the area server is then responsible for transferring the message to the recipients' MTAs within its area.

Messages from an area that are addressed to recipients at more than one MTA in another area are routed through the area server in the other area. Messages from an area that are addressed to recipients located at a single MTA in another area are sent directly to the destination MTA.

Messages that stay within the area are transferred directly between the MTAs in the area as usual.

Note

Messages are sent out of an area from the individual MTAs and are not transferred through the area server for that area.

You could use an MTA set as an area server, and reduce the possibility of having a single point of failure. Using an MTA set spreads the incoming message load across more than one MTA.

Message transfer from the originator's MTA directly to the destination MTA is more efficient, so take advantage of direct transfer where possible.

If you choose to have more than one area name in an MTA name, that is, you have a hierarchy of areas, each MTA is identified by a sequence of area names and an MTA name. The first area name represents the highest level of your geographical hierarchy, the second represents the next level, and so on.

If you decide not to implement area routing, you can still name your MTAs including one or more area names, but there is no need to identify an area server and all the MTAs can communicate directly with each other. This allows the flexibility of implementing area routing at a later stage.

Figure 4–3 Areas and Area Servers

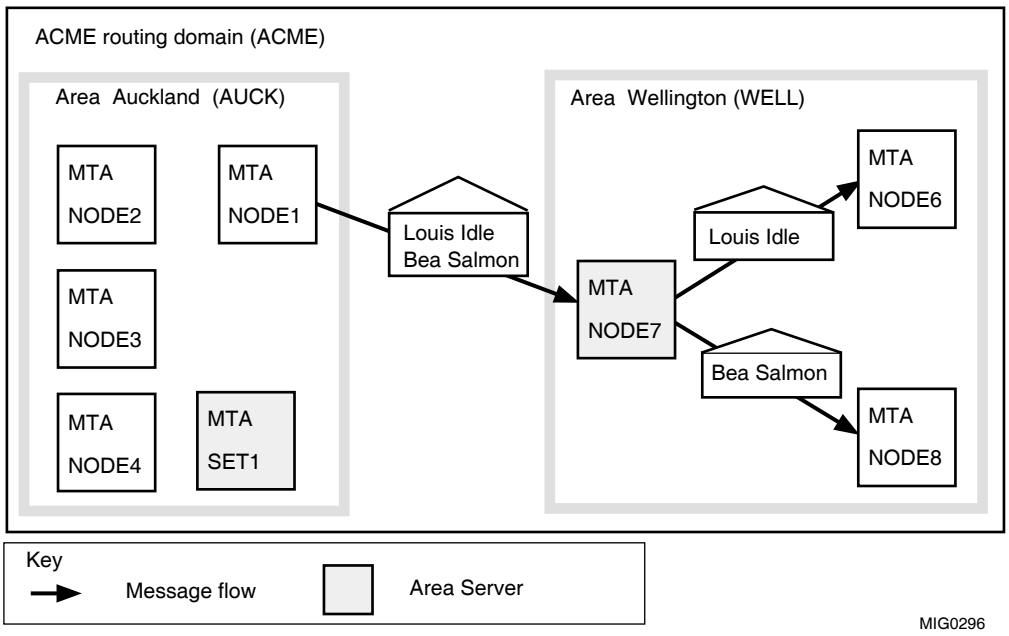


Figure 4–3 is an example of how the ACME Shoe Corporation could use area routing to divide the ACME routing domain into two areas, AUCK (Auckland) and WELL (Wellington). The figure shows how a single copy of a message is transferred from one area (AUCK) to two recipients in another area (WELL) through the area server for the WELL area. The area server then transfers the message to the two recipients. Note that the message is not sent out of the AUCK area through the area server for AUCK; it is sent directly to the area server for WELL from the originator's MTA. The area names, AUCK and WELL, make up part of the hierarchical MTA name.

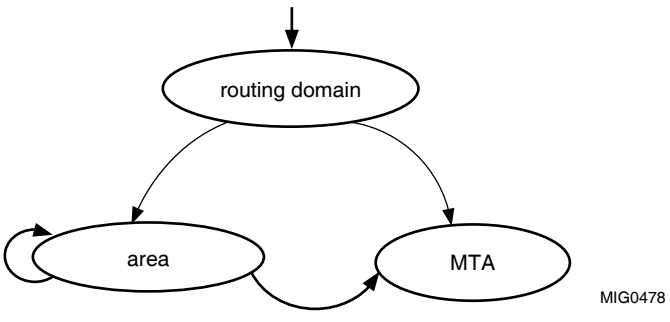
When you have decided how many of the areas in your geographical hierarchy you are going to represent in your routing domain, you need to assign a name to each area.

You can now plan names for the individual MTAs and MTA sets within each area.

4.3 Naming MTAs

A full MTA name has the hierarchical structure shown in Figure 4–4. This is how the MTA name is represented in the directory. Each arrow in the figure represents the permitted relationship between directory entries. For example, an MTA entry can be subordinate to the routing domain entry, or an area entry. Where an arrow loops back on itself, a hierarchy of multiple entries of the same type can be used.

Figure 4–4 Structure of an MTA Name



The structure of an MTA name comprises:

- The name of the routing domain to which the MTA belongs.

This name is common to all MTAs in your routing domain. In the case of the ACME Shoe Corporation, the routing domain name is ACME.

- A unique name that identifies an MTA within a routing domain. This is the MTA name and comprises:
  - Optionally, an area name or sequence of area names.

    You will find it easier if you define one area name from the start. This allows the flexibility of implementing area routing at a later stage without the need to rename the MTAs in the network. However, you do not need to plan an area name if you never intend to implement area routing.

    Area names are stored in the directory as Area entities, entities of the MTS module.
  - An MTA name.

    Each MTA name must be unique within the routing domain or area. It is a good idea to include the name of the node where the MTA is installed in the MTA name. It is also advisable to keep MTA names fairly short, thus limiting the number of characters that you have to type when you use the MTA name.

    MTA names are stored in the directory as MTA entities, entities of the MTS module.

You specify the routing domain name and MTA name when you install and set up each MTA.

As an example, the ACME Shoe Corporation has chosen to have two areas; one to represent Auckland and one to represent Wellington. ACME has chosen the following names for their MTAs, which include one area name for each MTA name:

AUCK.MTA-NODE1
AUCK.MTA-NODE2
AUCK.MTA-NODE3
AUCK.MTA-NODE4
AUCK.MTA-NODE11
AUCK.MTA-NODE12
AUCK.MTA-SET1
WELL.MTA-NODE6
WELL.MTA-NODE7
WELL.MTA-NODE8

The area names are AUCK and WELL, and the MTA names are based on the names of the nodes where the individual MTAs are installed. In this example, it is assumed that there is an expensive communications link between the Auckland and Wellington sites. It is therefore sensible to operate the two sites as areas to minimize the message traffic over the link. ACME identifies AUCK.MTA-SET1 and WELL.MTA-NODE7 to be the area servers for each of the areas AUCK and WELL respectively.

In this example, the area server AUCK.MTA-SET1 is an MTA set with members AUCK.MTA-NODE11 and AUCK.MTA-NODE12; Figure 4–3 shows the area server AUCK.MTA-SET1 serving the AUCK area.

Each MTA within your routing domain requires:

A name
A Presentation address
A password

The MTA name and password are provided by you when an individual MTA is set up. The Presentation address used by the MTA is set up automatically. The MTA password is used by the MTA when accepting a connection from a Peer MTA within its own routing domain. The password is also used when the MTA accesses routing information in the directory.

4.4 Naming Agents

This section describes what you need to plan for each type of Agent in your routing domain.

Agents are those applications that connect to the MTA and use its services, for example, User Agents, Gateways, and Message Stores. The Agents that connect to an MTA are either registered Agents or unregistered Agents of that MTA.

A registered Agent of the MTA is represented by an Agent entity and serves one or more O/R addresses. Typically, Gateways, User Agents serving a user community, and Message Stores are registered Agents. An unregistered Agent of the MTA serves only one O/R address and identifies itself to its MTA using that O/R address. An MTA has less information about its unregistered Agents than about its registered Agents. This is because unregistered Agents are not represented by Agent entities.

4.4.1 Registered Agents

Details of registered Agents of an MTA are held locally at the MTA to which the Agent connects. These details are held in an Agent entity, an entity of the MTA module.

For each registered Agent you require:

A name
Type of interface

You might also need to specify a password. Refer to the Agent documentation to determine whether a password is required.

Registered Agents that are Gateways can use either the XAPI interface or the Shared File interface to communicate with the MTA. All other Agents use the XAPI interface.

Contact the person responsible for managing the Gateway to obtain the information that you need about the Gateway.

4.4.1.1 XAPI Interface

The **XAPI interface** is based on the MAILbus 400 Application Program Interface (API). The XAPI interface can be used by any type of registered Agent: Gateway, User Agent or Message Store.

The MAILbus 400 API is a set of callable routines that you can use to build Agent applications that connect to an MTA and use its services. The applications that you build can be User Agents, Message Stores or Gateways. The User Agents and Gateways you build with the MAILbus 400 API can conform to the 1984 MHS Standards, the 1992 MHS Standards, or both. Message Stores were introduced in the 1988 X.400 recommendations and standards, and therefore must conform to the 1988 MHS Standards.

Where the Agent uses the XAPI interface, you might also need an invocation file name. The MTA uses the invocation file to prompt the Agent to connect to the MTA when the MTA has messages for the Agent. Contact the person responsible for managing the Agent to find out whether or not an invocation file is used by the Agent.

4.4.1.2 Shared File Interface

The **Shared File interface** is used by Gateways only. The Shared File 1984 interface is for Gateways that support only messages that are based on the 1984 MHS Standards, for example, Gateways developed for the Retix® OpenServer 400™. The Shared File 1992 interface is for Gateways that support the Shared File 1992, or 1988 interface, for example, Gateways developed for the ISOPLEX 800® MTA.

The Shared File interface allows a Gateway to pass a message to an MTA by placing it in an Input queue at the MTA, in the form of a file. The MTA reads the file, processes it, and deletes it from the queue afterwards. When an MTA passes a message to such an Agent, the MTA places the message in an Output queue in the form of a file. From there, the Agent reads the file, processes it, and deletes it from the queue afterwards.

If you want to use the MAILbus 400 MTA with a Gateway developed for use with the Retix ® OpenServer 400™, or an Access Unit developed for use with the ISOPLEX 800™ MTA, refer to Chapter 11. These Gateways and Access Units use the Shared File 1984 or the Shared File 1992 interface depending on their capability.

4.4.2 Unregistered Agents

User Agents that are unregistered Agents are identified by the O/R address of the individual that they serve. O/R addresses and passwords for unregistered Agents are stored in the directory. Chapter 5 describes how to allocate O/R addresses to individuals, including an individual served by an unregistered Agent.

4.5 Naming Other Routing Domains

Other routing domains can be either X.400 routing domains, or routing domains using non-X.400 messaging products that the MAILbus 400 MTA connects to through a Gateway. An example of a routing domain based on non-X.400 messaging products is the Internet network. The MAILbus 400 MTA can connect to an Internet network through the MAILbus 400 SMTP Gateway.

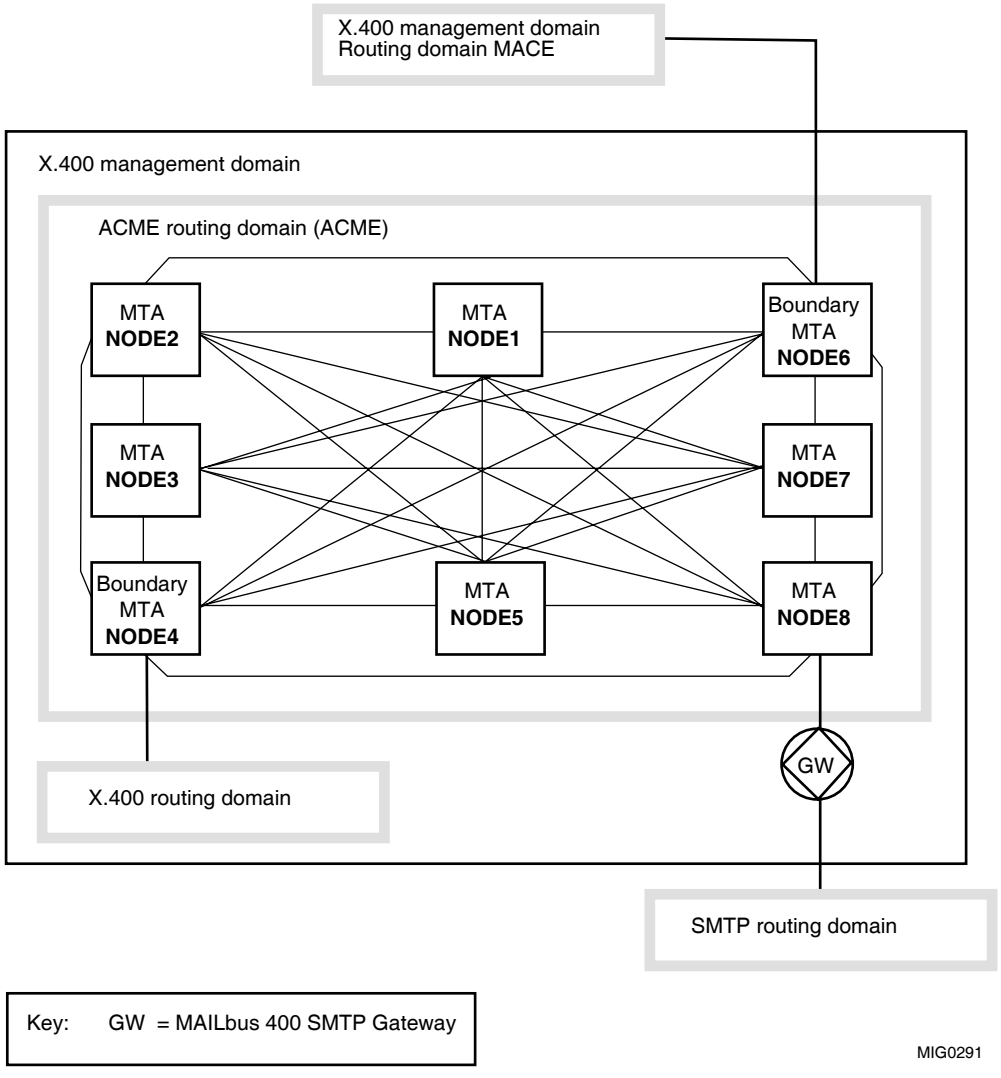
MTAs within your routing domain can connect to X.400 routing domains that are either within your X.400 management domain or part of another X.400 management domain.

If the other X.400 routing domains are within your X.400 management domain, they could be within your organization, but using another vendor's X.400 messaging product. Alternatively, they could be other routing domains made up of MAILbus 400 MTAs using different routing information.

All X.400 routing domains have GDIs. If another routing domain is in the same X.400 management domain as your routing domain, the GDIs for the routing domains are the same.

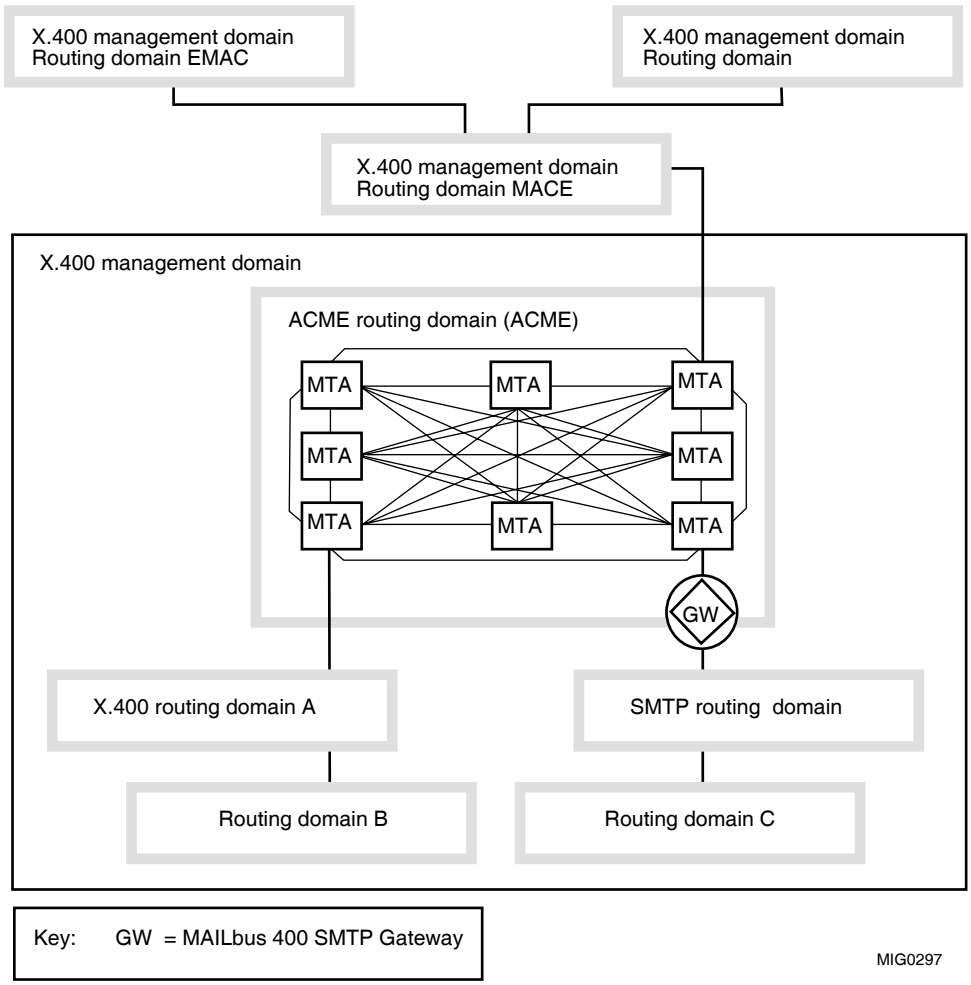
Figure 4–5 shows the layout of the ACME routing domain and other routing domains to which ACME MTAs connect. These are direct connections to other X.400 routing domains and connections through Gateways.

Figure 4–5 The ACME Routing Domain Within its X.400 Management Domain



MTAs can also transfer messages to other routing domains indirectly through an intermediate routing domain. Figure 4–6 shows the layout of the ACME routing domain and other routing domains, including the intermediate routing domains.

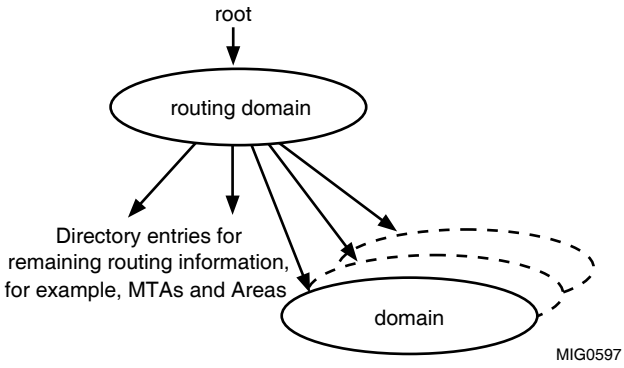
Figure 4–6 ACME Connections Through Intermediate Routing Domains



The connection to each other routing domain that MTAs within your routing domain connect to is represented in the directory as a Domain entity, an entity of the MTS module, and is identified by a name that you allocate. Routing information, which specifies how MTAs within your routing domain route

messages to individuals within these other routing domains, is also held in the directory. This routing information is described in detail in Chapter 5. Figure 4–7 shows the entries for other routing domains as domain entries subordinate to your routing domain entry in the directory. You also need to add information in the directory that indicates whether the routing domain is part of another X.400 management domain.

Figure 4–7 Domain Entries in the Directory



Details of how a boundary MTA in your routing domain communicates with MTAs in other routing domains are held locally in Peer MTA entities (see Section 4.6). With reference to Figure 4–5, there are two boundary MTAs, MTA-NODE4 and MTA-NODE6. Section 4.6 describes in detail the information that is required for Peer MTAs.

Details of how MTAs in your routing domain connect to Gateways are held in Agent entities. This is described in Section 4.4.

4.6 Peer MTA Entities

Peer MTA entities are entities of the MTA module and are created either automatically or manually. The entities are referred to as automatically-configured or manually-configured Peer MTA entities (Section 4.6.1 and Section 4.6.2 respectively).

4.6.1 Automatically-Configured Peer MTA Entities

MAILbus 400 MTAs create Peer MTA entities automatically when they make connections to peer MAILbus 400 MTAs within the same routing domain. As an example and with reference to Figure 4–5, when MTA NODE4 connects to MTA NODE3. Automatically-configured Peer MTA entities hold management information about the peer MTA connection.

When a MAILbus 400 MTA connects to a peer MTA in the same routing domain, it obtains the information it requires about the peer MTA from the information in the directory relating to that peer MTA. The information held in the directory includes name, password and address (see Section 4.3).

The information held in automatically-configured Peer MTA entities is statistical information about an MTA's connection to the peer MTA.

There is no management information that you need to set up for automatically-configured Peer MTA entities.

4.6.2 Manually-Configured Peer MTA Entities

You manually create one or more Peer MTA entities at each boundary MTA in your routing domain to represent Peer MTAs in other routing domains. A Peer MTA entity that is created manually holds information about a peer MTA in another routing domain. This information is required to manage connections between the boundary MTA and the peer MTA. Note that a boundary MTA can potentially communicate with more than one peer MTA, in one or more different routing domains. In this case, you create a Peer MTA entity for each peer MTA with which the boundary MTA intends to communicate.

If you use an MTA set as a boundary MTA, you can nominate each member of the MTA set to connect to one or more peer MTAs in the other routing domain. However, you are recommended to set up all MTAs in the set identically, as described in Section 4.2.1. Each MTA in the MTA set should be able to connect to the same peer MTA, or number of peer MTAs.

When a MAILbus 400 MTA connects to a peer MTA in another routing domain, it obtains the information it requires about the peer MTA from a Peer MTA entity. You have to set up the information in the Peer MTA entity as part of setting up your messaging system.

Contact the person responsible for managing the peer MTA to obtain some of the required information about the peer MTA. The information required for a Peer MTA entity that is manually configured comprises:

- A name, which associates the Peer MTA entity with the connection to the peer MTA.

This could be the name of the peer MTA, or it could be another name that you have chosen.

- The type of X.400 connection used between the peer MTA and the boundary MTA.

  This is called the **application context** and can be one of:

  – MTS Transfer Protocol 1984.

    This indicates that the peer MTA only sends and accepts 1984 X.400 messages using OSI Session connections. In this case, the boundary MTA will downgrade messages (adapt 1992 X.400 messages to a 1984 X.400 system) before it transfers them out of the routing domain.

  – MTS Transfer Protocol.

    This indicates that the peer MTA sends and accepts 1992 X.400 messages, using OSI Session connections. In this case, the peer MTA transfers 1992 X.400 messages.

  – MTS Transfer.

    This indicates that the peer MTA fully supports 1992 MHS Standards and uses OSI ACSE associations. This is the application context used between MAILbus 400 MTAs.

- Either a **Presentation address** or a **Session address** for the peer MTA.

  The type of address depends on the application context used by the peer MTA.

  If the peer MTA uses an application context of MTS Transfer, you provide its Presentation address. Otherwise, you provide its Session address. Note that if the peer MTA intends to use a Session address, a Session address must also be provided for the boundary MTA. The Session address for the boundary MTA is determined automatically, if requested, when the boundary MTA is set up.

- The type of **Dialogue mode** used between the peer MTA and the boundary MTA.

  This can be one of:

  – Monologue

    The MTA uses a single association to either send or receive MPDUs, but not both at the same time.

  – Two-way Alternate

    The MTA uses a single association to alternately send and receive MPDUs.

If you do not specify a value for the dialogue mode, the MTA uses monologue dialogue mode.

- The name of the other routing domain.

  This is a name that you assign to the routing domain where the peer MTA is located. It is used by a boundary MTA to determine which Peer MTA entity, or entities, to use for information about potential connections to the named routing domain. It is also the identifier for the domain entry in the directory that represents the connection to the routing domain to which the peer MTA belongs.

- Authentication information, when required, for communications between the boundary MTA and the peer MTA.

- Transport service. (Tru64 UNIX)

  This is the transport service that you want a boundary MTA to use when making connections to the peer MTA, and can be either the OSI Transport Service or TCP/IP Transport Service. This option is only applicable to the MAILbus 400 MTA for Tru64 UNIX.

  If you do not specify a transport service, the boundary MTA uses the transport service defined for communication within the routing domain.

- Transport template names.

  This is a list of names of the Transport templates that are used by the boundary MTA when making connections to the peer MTA using the HP DECnet-Plus Transport Service (Tru64 UNIX) or HP DECnet-Plus and TCP/IP Transport Service (OpenVMS).

  A Transport template is held in a Template entity, which is an entity of the OSI Transport module. The identifier of a Template entity is used as the template name defined for the Peer MTA entity. *HP MAILbus 400 MTA Tuning and Problem Solving* explains the MAILbus 400 MTA's use of Template entities in more detail.

  For OpenVMS, use the order of the Transport Template names to specify the transport service that you want the boundary MTA to use when making connections to the peer MTA.

  If you do not specify any Transport template names for the peer MTA, the boundary MTA uses the Transport template defined for communication within the routing domain.

  When you define a Template entity with a Network Service attribute value of CONS, a default X.25 Template is automatically assigned. Check that the network described by this X.25 Template is accessible to the boundary MTA.

4.7 Summary of Planning MTA and Routing Domain Names

You have now planned names for your routing domain, the MTAs that make up the routing domain, and the routing domains to which your routing domain connects.

Use Table 4–1 to complete the details of the values that you have planned for the names of MTAs and the names of other routing domains. MTA names and routing domain names are entered in the directory when you set up the routing information for your routing domain. Part III of this book describes how to set up your routing domain.

Table 4–1 Worksheet for Your Routing Domain

Attribute	Value	Comments
Routing Domain Name[1]	/MTS=*routing-domain-name*	*routing-domain-name* is the value for the relative distinguished name (RDN) given to the routing domain (no length constraint, printable string[2]).
Password[1]	_____	Password (up to 62 characters, printable string[2]).
Global Domain Identifiers[1]	_____	List of one or more GDIs used within your routing domain (see Section 4.1.1).
For Each Area in the Routing Domain		
Identifier	_____	Part of MTA name (printable string[2], see MTA identifier). Area identifiers are optional, but at least one is recommended.
Server MTA	_____	Name of the MTA serving the area (optional).

[1]If you have assigned values to these attributes when reading *MAILbus 400 Getting Started* use the same values.

[2]Refer to *HP MAILbus 400 MTA Tuning and Problem Solving* or the *MTS Module Online Help* for details of individual character sets.

(continued on next page)

Table 4–1 (Cont.) Worksheet for Your Routing Domain

Attribute	Value	Comments
For Each MTA in the Routing Domain		
Identifier[1]	_ _ _ _ _ _	MTA name, includes any Area identifiers, where applicable. Overall the MTA name can be no more than 32 characters (printable string[2]). The length includes periods (.), used to separate the different elements of the identifier; for example AUCK.MTA-NODE1.
Password[1]	_ _ _ _ _ _	Password (up to 62 characters, printable string[2]).
For Each MTA Set in the Routing Domain		
Identifier	_ _ _ _ _ _	Name that identifies the MTA set (see MTA Identifier).
Members	_ _ _ _ _ _	List of MTA names (see MTA identifier).
For Each Routing Domain to which MTAs in your Routing Domain Connect		
Identifier[1,3]	_ _ _ _ _ _	Name given to the other routing domain (maximum of 128 characters, printable string[2]).

[1]If you have assigned values to these attributes when reading *MAILbus 400 Getting Started* use the same values.

[2]Refer to *HP MAILbus 400 MTA Tuning and Problem Solving* or the *MTS Module Online Help* for details of individual character sets.

[3]This attribute must match Peer Domain (see Table 4–2).

(continued on next page)

Table 4–1 (Cont.) Worksheet for Your Routing Domain

Attribute	Value	Comments
For Each Routing Domain to which MTAs in your Routing Domain Connect		
Different CCITT Domain[1,4]	_____	An indication of whether or not the named routing domain is in another X.400 management domain (either TRUE or FALSE).
Password[1,5]	_____	Password (up to 62 characters, printable string[2]).

[1]If you have assigned values to these attributes when reading *MAILbus 400 Getting Started* use the same values.

[2]Refer to *HP MAILbus 400 MTA Tuning and Problem Solving* or the *MTS Module Online Help* for details of individual character sets.

[4]You can optionally include warning text within messages that are received from another X.400 management domain. Part II of *HP MAILbus 400 MTA Tuning and Problem Solving* explains how you tune the MTA to add warning text to messages. You can also add permission information to O/R addresses to prevent a user with a particular O/R address exchanging mail across an X.400 management domain boundary (see Section 5.13).

[5]The password might be required for a routing domain served by a Gateway. Refer to the Gateway documentation for more information.

Chapter 7 provides examples of how to create these entries using the ACME Shoe Corporation as an example. Part III of this book describes how you set up your routing domain using this information.

4.8 Summary of Planning Peer MTA Entities

Use Table 4–2 to fill in details of the planned connections between a boundary MTA and a peer MTA in another routing domain. These values are entered as Peer MTA entities when you set up the boundary MTA.

If you have already set up peer MTA entries as part of reading *MAILbus 400 Getting Started*, you do not need to complete this table.

Table 4–2 Worksheet for Peer MTA Entity

Attribute	Value	Comments
For Each Peer MTA to which an MTA in Your Routing Domain Connects		

(continued on next page)

Table 4–2 (Cont.) Worksheet for Peer MTA Entity

Attribute	Value	Comments
For Each Peer MTA to which an MTA in Your Routing Domain Connects		
MTA Identifier	_ _ _ _ _	Name of boundary MTA.
Peer MTA entity name	_ _ _ _ _	Name given to Peer MTA entity. (No length constraint, alphanumeric and can include hyphens. The leading character must be alphabetic.)
Application Context	_ _ _ _ _	The type of connection used between the boundary MTA and the peer MTA (see Section 4.6).
Presentation Address[1]	_ _ _ _ _	The address of the peer MTA, required for an Application Context of MTS Transfer.
Session Address[1]	_ _ _ _ _	The address of the peer MTA, required for an Application Context of MTS Transfer Protocol 1984 or MTS Transfer Protocol.
Dialogue Mode	_ _ _ _ _	The type of dialogue mode used between the boundary MTA and the peer MTA. This can be either "MONOLOGUE" or "TWO WAY ALTERNATE". If you do not specify a value for the dialogue mode, the MTA uses monologue dialogue mode. *HP MAILbus 400 MTA Tuning and Problem Solving* describes the MTA's use of the dialogue mode in detail.
Peer Domain[2]	_ _ _ _ _	Name given to the routing domain where the peer MTA is located (up to 128 characters, printable string[3]).
Peer Name	_ _ _ _ _	For authentication, provided by the manager of the peer MTA, when required (up to 32 characters, IA5 graphic subset)[3].

[1]Either a Presentation or a Session address, according to the application context used.

[2]There must be a routing domain with this identifier defined in Table 4–1.

[3]Refer to *HP MAILbus 400 MTA Tuning and Problem Solving* or the *MTS Module Online Help* for details of individual character sets.

(continued on next page)

Table 4–2 (Cont.) Worksheet for Peer MTA Entity

Attribute	Value	Comments
For Each Peer MTA to which an MTA in Your Routing Domain Connects		
Peer Password	_____	For authentication, provided by the manager of the peer MTA, when required. Case sensitive, between 1 and 64 characters (IA5 graphic subset[3]), unless application context is MTS Transfer, in which case, 62 characters (IA5 graphic subset or Octet string)[3].
Local Name	_____	For authentication, provided by the manager of the boundary MTA and also given to the manager of the peer MTA (IA5 graphic subset[3], up to 32 characters).
Local Password	_____	For authentication, provided by the manager of the boundary MTA and also given to the manager of the peer MTA. Case sensitive, between 1 and 64 characters (IA5 graphic subset[3]), unless application context is MTS Transfer, in which case, 62 characters (IA5 graphic subset or Octet string)[3].
Transport Service Option (Tru64 UNIX)	_____	The type of transport service that the boundary MTA is to use when communicating with the peer MTA. This can be either, or both, "OSI" or "TCPIP". Only required if the peer MTA uses a different transport service from that specified for the boundary MTA. You can specify one or both transport services, in the preferred order of use. *HP MAILbus 400 MTA Tuning and Problem Solving* describes the MTA's use of the transport service in detail.

[3]Refer to *HP MAILbus 400 MTA Tuning and Problem Solving* or the *MTS Module Online Help* for details of individual character sets.

(continued on next page)

Table 4–2 (Cont.) Worksheet for Peer MTA Entity

Attribute	Value	Comments
For Each Peer MTA to which an MTA in Your Routing Domain Connects		
Template Name	_ _ _ _ _ _	The names of the Transport layer templates that the boundary MTA uses when communicating with the peer MTA using OSI Transport. Only required if the peer MTA uses transport characteristics that are different from those set up for the boundary MTA. *HP MAILbus 400 MTA Tuning and Problem Solving* describes the MTA's use of Transport templates in detail. Transport Template entities are explained in detail in the appropriate DECnet-Plus documentation.

Chapter 7 provides examples of how to create these entities using the ACME Shoe Corporation as an example. Part III describes how you set up your routing domain and create Peer MTA entities using the information that you have planned.

4.9 Summary of Planning Agent Entities

Use Table 4–3 to fill in details of the values that you have planned for connections between an MTA and the registered Agents that it serves. These values are entered in Agent entities at the MTA to which the Agent is to connect. Note that for MAILbus 400 Agents such as the HP MAILbus 400 SMTP Gateway or the MAILbus 400 Message Store, this information is created when the Agent is set up. Refer to the product documentation for more information.

If you have already set up your Agent entries as part of reading *MAILbus 400 Getting Started*, you do not need to complete this table.

Table 4–3 Worksheet for Agent Entity

Attribute	Value	Comments
For Each Registered Agent in the Routing Domain		
MTA Name	_____	Name of MTA to which the Agent connects.
Identifier	_____	A name identifying the Agent. For an Agent using the XAPI interface, this is 128 characters, alphanumeric and can include hyphens. For an Agent using the Shared File 1984 or the Shared File 1992 interface, this is 8 characters, alphanumeric and can include hyphens. In both cases, the leading character must be alphabetic. Note that you cannot use spaces in the Agent name.
Password	_____	Where required by the Agent, case sensitive, between 1 and 62 characters (printable string[1]).
Type	_____	Describes the type of interface that the Agent uses (XAPI, SHARED FILE 1984 or SHARED FILE 1992). Specify this attribute when you create the Agent.

[1]Refer to *HP MAILbus 400 MTA Tuning and Problem Solving* or the *MTS Module Online Help* for the printable string character set.

(continued on next page)

Table 4–3 (Cont.) Worksheet for Agent Entity

Attribute	Value	Comments
For Each Registered Agent in the Routing Domain		
Invocation Filename	_____	Where required by the Agent, a filename. (For Agents supporting the XAPI interface.)

Chapter 11 describes how to make sure that an Agent can connect to an MTA, when the MTA and Agent are installed and set up.

4.10 What to Do Next

You have now completed naming MTAs, Agents, and routing domains. Read Chapter 5, which describes planning O/R addresses and routing information for individuals in your routing domain and in other routing domains.

5

O/R Addresses and Routing Information

This chapter describes what you need to consider when planning O/R addresses and routing information for individuals within your routing domain. It also describes the O/R addresses that you need to define for individuals that you want to be able to communicate with in other routing domains. These other routing domains can be X.400 routing domains or routing domains served by Gateways.

5.1 Addressing in an X.400 MHS

In order to transfer messages, MTAs need address information for the originator and for each recipient of a message. According to the 1992 MHS Standards, users are identified using originator/recipient names (**O/R names**). An O/R name identifies a user as the originator or recipient of a message and contains either one or both of the following:

- The user's originator/recipient address (O/R address)

  O/R addresses are described in detail in Section 5.1.1. The MAILbus 400 MTA only transfers a message if the originator and recipients are specified with O/R names that contain O/R addresses. The use of O/R addresses enables the MAILbus 400 MTA to interwork with 1984 X.400 messaging systems, because such systems support only O/R addresses. Note that in 1984 messaging systems, O/R addresses are referred to as O/R names.

- The user's **Directory name**

  The Directory name is a specific way of uniquely identifying a user, but is as yet not widely supported in X.400 implementations. The MAILbus 400 MTA does not support transfer of messages whose originator or recipients are specified with O/R names that contain only Directory names.

5.1.1 O/R Addresses

Each O/R address attribute consists of an attribute type and one or more values. The hierarchical structure of O/R addresses enables them to be globally unique. The hierarchy of O/R address attributes and the uniqueness of O/R addresses on a global scale are safeguarded by a number of authorities and by the organizations participating in the global messaging system. Section 5.1.2 explains this in more detail.

There are four forms of O/R address:

- **Mnemonic** (see Section 5.1.1.1)
- **Numeric** (see Section 5.1.1.2)
- **Postal** (see Section 5.1.1.3)
- **Terminal** (see Section 5.1.1.4)

This chapter concentrates on the mnemonic O/R address form, which is recommended in the 1992 MHS Standards for naming individuals who use electronic messaging. Mnemonic O/R addresses are considered to be the most friendly of the O/R address forms. Postal and terminal O/R addresses accommodate more traditional means of message transfer, while numeric O/R addresses enable organizations to continue using identification schemes based on numbers.

5.1.1.1 Mnemonic O/R Addresses

A mnemonic O/R address is a set of attributes whose values reflect the geographical or organizational hierarchy in which a user is located. Each attribute represents one level in the hierarchy. A mnemonic O/R address specifies, for example, the country in which a user works or lives, the organization in which a user works, and the user's name. It is a form of O/R address that is easy to use because it is a combination of words or logical abbreviations.

You can compare the mnemonic O/R address to the mail address used when sending a letter. To send a letter to someone, you have to specify various address items on the envelope. These could be, for example, the name of the person, the house, the street, the city, and the country. A mnemonic O/R address is based on the same principle; its address items are the different O/R address attribute values.

The 1992 MHS Standards suggest the use of mnemonic O/R addresses for electronic message transfer.

5.1.1.2 Numeric O/R Addresses

Numeric O/R addresses identify individual users by means of numeric user identifiers. Although this O/R address form is not as easy to use as the mnemonic one, you may prefer numeric O/R addresses for members of your organization if your organization has already implemented a system of identifying its members by numbers (for example, employee payroll or salary numbers).

The numeric O/R address is described in this chapter as an alternative form of O/R address for naming individuals. You will need to decide whether you are going to use the numeric or mnemonic form of O/R address to name individuals in your routing domain.

5.1.1.3 Postal O/R Addresses

Postal O/R addresses are used in messaging systems that are based on the delivery of mail by a postal service. They are used primarily for message exchange with physical delivery Access Units. A postal O/R address specifies the delivery system through which a user's paper mail is delivered. It also specifies a user's postal address.

The postal form of O/R address is described in Appendix A.

5.1.1.4 Terminal O/R Addresses

Terminal O/R addresses identify participants in a messaging system by the location of their terminals within the network. Terminal O/R addresses are used primarily for message exchange with telex, teletex, and facsimile machines through suitable Access Units.

The telematic form of O/R address is described in Appendix A.

5.1.2 Ensuring the Uniqueness of O/R Addresses

Each O/R address in a global messaging system must uniquely identify a user. Therefore, each O/R address must be globally unambiguous. As it would be impractical for one central authority to coordinate the assignment of unique O/R addresses, the 1984 and 1992 MHS Standards define a hierarchy of registration authorities.

A **registration authority** is a body that has been given responsibility for assigning unique values to the countries, bodies, and organizations participating in a global messaging system. Examples of registration authorities are **administration management domains (ADMDs)** and **private management domains (PRMDs)**:

- ADMD

  An ADMD is a messaging domain that is operated by a CCITT administration. In many countries a CCITT administration is the country's Post, Telegraph, and Telephone authority (PTT), while in others, ADMDs are run by privately-owned companies. A country can have several ADMDs which provide public messaging services within that country.

  As ADMDs are connected on an international level, they also provide international messaging services. Organizations can subscribe to an ADMD's communications network and use its services.

- PRMD

  A PRMD is a messaging domain that is not controlled by a CCITT administration, but is owned and managed by a private organization, for example, a company.

ADMDs and PRMDs are called **X.400 management domains**. In the context of electronic message transfer, these domains assign unique O/R address attribute values to their subordinate authorities or organizations. These are the attribute values at the top of the O/R address hierarchy. The attribute values further down in the O/R address hierarchy reflect the hierarchical structure of your organization. They are decided by you and are determined by an addressing scheme that you plan.

If each registration authority assigns a unique attribute value to each subordinate authority, and if the individual organizations assign unique attribute values to their users, each O/R address in the global messaging system is unique.

5.2 Why You Need O/R Addresses

MAILbus 400 MTAs use O/R addresses to route messages. O/R addresses and associated routing information are held in the directory and are available to all the MTAs in a routing domain.

The attributes of an O/R address are ordered hierarchically. Figure 5–1 indicates the hierarchy of the mnemonic form of O/R address.

Figure 5–1 How Mnemonic O/R Addresses are Constructed

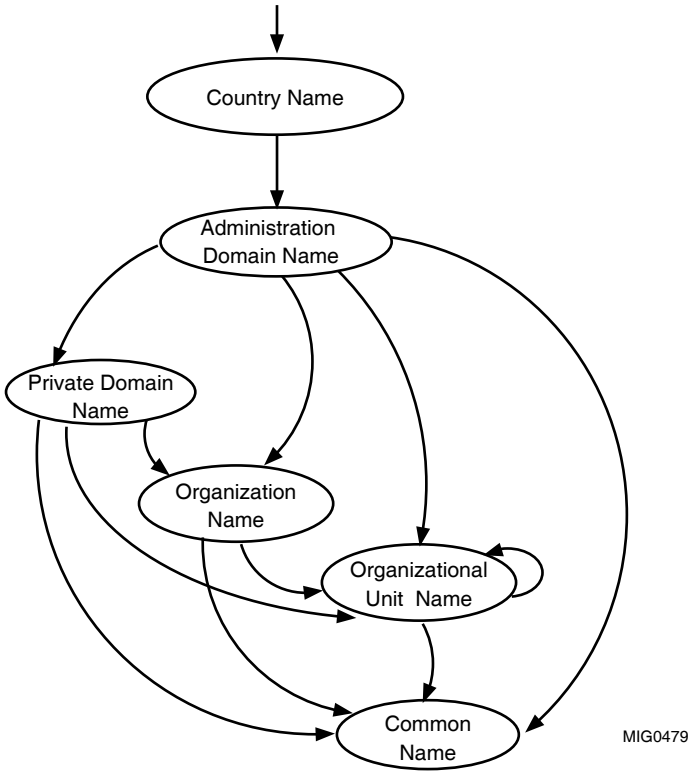
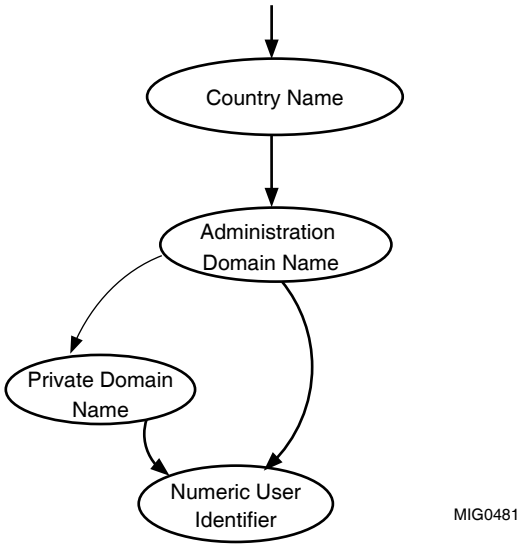


Figure 5–2 shows the hierarchy of the numeric form of O/R address. Each arrow in the figure represents the permitted relationship between directory entries.

Figure 5–2 How Numeric O/R Addresses are Constructed



Similar descriptions of the terminal and postal O/R addresses are shown in Appendix A. Mnemonic and numeric O/R address attributes are described in Section 5.3. The hierarchical ordering of O/R address attributes is the basis for storing O/R addresses in the directory. MTAs can then use these O/R addresses for routing.

Routing information defines the action that an MTA takes when routing messages to a particular O/R address. The action can be:

- Deliver the message to an MTA and Agent that serve the O/R address.
- Transfer the message to another routing domain.
- Redirect the message to another O/R address.
- Do not deliver the message.

A partial O/R address is a series of leading attributes of an O/R address. Routing information defined for a partial O/R address applies to all more complete O/R addresses that contain the partial O/R address and that have no other routing information defined in their O/R address hierarchy. This allows you to represent once, for the partial O/R address, routing information that is common to a group of users.

There is additional information that can be represented in the directory as attributes of the O/R address entries, for example, a Personal Name attribute. This information is described later in this chapter.

Before planning the routing information for your O/R address entries, you need to plan values for the O/R address attributes that you want to use to name individuals in your routing domain. Section 5.3 describes the attributes of O/R addresses.

5.3 O/R Address Attributes

For each form of O/R address, the 1992 MHS Standards define some O/R address attributes as mandatory and others as conditional. Each attribute has a type and zero or more values. The 1992 MHS Standards allows teletex values for some O/R address attributes. If you intend to interwork solely with systems based on the 1984 MHS Standards, you might want to assign printable string values to O/R address attributes rather than teletex. However, by using aliases, you can use teletex strings for communication within your routing domain and printable strings for communication to systems based on the 1984 MHS standards. Section 5.6 describes aliases in more detail.

Table 5–1 shows the mandatory and conditional attribute types of the mnemonic and numeric O/R address forms as defined by the 1992 MHS Standards. Similar descriptions of the terminal and postal O/R addresses can be found in Appendix A. Table 5–1 also indicates the labels that you use when entering O/R addresses and attributes used by the MAILbus 400 MTA in the directory.

Table 5–1 O/R Address Attributes

Attribute	Label	Used by MTA for Routing	String Length and Character Set	Mandatory (M) or Conditional (C)
Attributes Common to Mnemonic and Numeric O/R Addresses				
Country Name	C	Yes	2 (alphabetic) or 3 (numeric)	M
Administration Domain Name	A	Yes	16 (numeric or printable)	M
Private Domain Name	P	Yes	16 (numeric or printable)	C
Domain Defined 1 Domain Defined 2 Domain Defined 3 Domain Defined 4	DDA:	No	8 (type): 128 (value) (teletex or printable)[1]	C
Attributes Specific to Mnemonic O/R Addresses				
Organization Name[2]	O	Yes	64 (teletex or printable)	C
Organizational Unit 1	OU1	Yes	32 (teletex or printable)	C
Organizational Unit 2	OU2	Yes	32 (teletex or printable)	C
Organizational Unit 3	OU3	Yes	32 (teletex or printable)	C
Organizational Unit 4	OU4	Yes	32 (teletex or printable)	C
Common Name	CN	Yes	64 (teletex or printable)	C

[1]To enter a DDA Type in the directory that contains an equals sign "=", you need to specify an escape sequence. For the MAILbus 400 MTA this is done by doubling the equals sign; for example, DDA type of fax=fax22 is entered as: DDA:fax= =fax22=+44 000 000000

[2]Recommended if Organizational Units are used.

(continued on next page)

Table 5–1 (Cont.) O/R Address Attributes

Attribute	Label	Used by MTA for Routing	String Length and Character Set	Mandatory (M) or Conditional (C)
Attributes Specific to Mnemonic O/R Addresses				
Personal Name comprising:	-	Yes[3]	-	C
Given Name	G	-	16 (teletex or printable)[4]	C
Initials	I	-	5 (teletex or printable)[4]	C
Surname	S	-	40 (teletex or printable)[4]	M[5]
Generation Qualifier	Q	-	3 (teletex or printable)[4]	C
Attributes Specific to Numeric O/R Addresses				
Numeric User ID	N-ID	Yes	32 (numeric)	M

[3]Personal names can be used for routing where there is no Common Name attribute specified on the message (see Section 5.3.1.7).

[4]When entering this attribute in an O/R address, you must use the printable string characters so that the message can be successfully downgraded (see *HP MAILbus 400 MTA Tuning and Problem Solving*). In all other cases where this attribute is used, for example, as part of a redirection O/R address, either a printable string or teletex string can be used.

[5]When a Personal name is used this attribute is mandatory.

Note that for the mnemonic form of O/R address, the 1992 MHS Standards state that at least one of the conditional attributes in Table 5–1 must be specified for O/R addresses.

The minimum number of attributes for a mnemonic O/R address is therefore:

- Country Name
- Administration Domain Name
- One conditional attribute, for example, Common Name

Note that if your organization is a Private Management Domain, you must also include a Private Domain Name.

The hierarchical ordering of the O/R address attributes enables a hierarchy of registration authorities to assign values to some attributes of the O/R address to ensure uniqueness. The values assigned by these registration authorities are the GDI attributes.

5.3.1 O/R Address Attribute Descriptions

If you intend your MTAs to connect to MTAs in a routing domain in another X.400 management domain, there might be rules about the attributes that you can and cannot include in your O/R addresses. Find out about any such rules before you start to plan your O/R addresses. Do this by contacting your national standards organization for information. In the case of connecting your organization's routing domain to an Administration Management Domain (ADMD), the organization providing the ADMD service can give you this information.

If you have multiple values for any of the O/R address attributes, for example, the Common Name attribute, you can include the alternative values in the directory as aliases. Aliases are also used where you have multiple GDIs in your routing domain (see Section 5.6).

5.3.1.1 Country Name

The Country Name attribute identifies the country where your routing domain is located. Country Name is a mandatory attribute, so you must always include it in your naming scheme, even if you are not planning to connect to routing domains in other countries.

The International Standard ISO 3166 defines one or more two-character strings to represent each country name. CCITT Recommendation X.121 defines one or more three-digit strings. Some countries are only represented by the ISO 3166 codes, but most are represented by both ISO and CCITT codes. Use any or all of the identifiers representing your country as the value(s) of the Country Name attribute. Appendix D contains a list of the ISO and CCITT country codes, for example, New Zealand is represented by NZ and 530.

If your routing domain spans more than one country, you can use any or all of the Country Name attribute values for each country that your routing domain is in. Consult each country's registration authority, the national standards organization, about the conditions under which each country code can be used with respect to the other X.400 O/R address attributes.

5.3.1.2 Administration Domain Name

The Administration Domain Name attribute identifies an ADMD within the country specified by the Country Name attribute. An ADMD is an X.400 management domain controlled by an organization providing an X.400 messaging service. Most countries have at least one ADMD and some have more than one. Administration Domain Name attributes are allocated on a national basis by a national standards body that allows the ADMD to operate the service. This service is usually operated by the country's Post, Telegraph, and Telephone (PTT) authority.

The regulations that apply when connecting to an ADMD are set by the organization providing the ADMD services and differ from country to country. Find out about the laws, regulations, and restrictions that apply to your organization. To do this, contact the organization(s) providing the ADMD service(s) for your country.

If you are implementing a small MTS configuration and never intend to have connections to any other X.400 management domain, you must still include the Administration Domain Name attribute in your naming scheme because it is a mandatory attribute. However, you should use a single 0 (zero) as its value in this case.

In some countries where there is more than one ADMD, the ADMDs have an agreement to relay messages from any Private Management Domain (PRMD) in the country that is connected to any of the ADMDs. If this convention is in use in your country, assign a single " " (space), as one of the values for the Administration Domain Name attribute in your naming scheme. Contact the organization providing the ADMD service to find out whether they have such an agreement.

If you plan to connect your routing domain to one or more ADMDs, you must register with each organization that provides the ADMD service to be assigned a value for the Administration Domain Name attribute applicable for that organization. The organization providing the ADMD service also provides the corresponding Country Name attribute where required. The name of that ADMD then becomes part of your users' O/R addresses.

If you expect to connect your routing domain to an ADMD in the future, try to use the name of the ADMD that you expect to connect to as the value for this attribute. This avoids the need to change O/R address attribute values in the future.

5.3.1.3 Private Domain Name

The Private Domain Name attribute identifies a PRMD within a country specified by the Country Name attribute or an ADMD specified by the Administration Domain Name attribute. A PRMD is an X.400 management domain.

The Private Domain Name attribute is required if:

- You plan to connect to an ADMD and the ADMD has naming regulations for all the PRMDs that connect to it.
- You plan to have a private network that is not connected to an ADMD, but possibly connected to other PRMDs, that is, you are using the single 0 (zero) convention for the Administration Domain Name attribute. In this case, the PRMD can be registered directly with the country through the national standards body.

If the PRMD is not connected to any other X.400 management domain, it need not be registered.

The combinations of Country Name, Administration Domain Name, and Private Domain Name must be unique among all the PRMDs that your organization is either directly or indirectly connected to. This combination of attributes is your GDI.

If you are connecting to an ADMD, it is the ADMD operator's responsibility to ensure that all PRMDs are identified uniquely. This is important where a number of ADMDs have an agreement to use the single space convention (see Section 5.3.1.2). In which case, the organization providing the ADMD service needs to ensure that all PRMD names are unique for all the PRMDs that take part in this agreement.

There are rules and regulations that apply to PRMDs that you must find out before assigning a value to this attribute. For example, some countries do not allow direct connections between PRMDs, or connections from a PRMD to more than one ADMD, if the PRMD is within an ADMD public network.

Most ADMD operators allow you to request a value for your PRMD. If this value is unique among the PRMDs that the ADMD serves, you can use it in your naming scheme. This is useful if you have connections to several ADMDs and want to use the same PRMD value for each set of O/R address attributes when connecting to these ADMDs.

5.3.1.4 Organization Name

The Organization Name attribute identifies an organization.

If you are connecting to another X.400 management domain through an ADMD, and you have allocated a PRMD name, this attribute follows the PRMD name.

You can use any value you want for this attribute, as long as it is unique within the X.400 management domain or country represented by the attribute immediately preceding this attribute in the hierarchy. Normally, you use the name of your organization as the value of this attribute.

If you are operating a private MTS and using a single 0 (zero) as an Administration Domain Name, you must choose an Organization Name that is unique within the country in which you are operating. In this case, you must apply to the national registration authority within the country for a value for this attribute.

Note that if you include an Organizational Unit Name attribute, the 1992 MHS Standards mandate that an Organization Name attribute is present in an O/R address. However, the 1984 MHS standards were more flexible and allowed you to omit the Organization Name attribute when including Organizational Unit Name attribute values. The MAILbus 400 MTA supports O/R addresses that have Organizational Unit Name attributes and no Organization Name attribute.

5.3.1.5 Organizational Unit Name

The Organizational Unit Name attribute identifies subdivisions of the organization denoted by the Organization Name attribute. You are advised to include an Organization Name attribute if you use Organizational Unit Names (see Section 5.3.1.4). Each O/R address can contain between zero and four Organizational Unit Name attributes.

The Organizational Unit Name attribute identifies each level of the organizational hierarchy that you have decided to include in your naming scheme. Use an Organizational Unit Name attribute to create the part of your naming scheme that represents your organization's hierarchy.

Organizational Unit Name attributes are ordered hierarchically, so that the first unit in the hierarchy represents a subdivision of the organization itself. For example, the values of ACME Shoe Corporation's first level Organizational Unit Name attribute represent the geographical divisions, Wellington and Auckland. The second unit in the hierarchy represents the subdivision of one of the first units. ACME does not have a second unit in its hierarchy.

You can choose any value for each Organizational Unit Name attribute as long as it is unique within the organization or organizational unit represented by the attribute immediately prededing it in the O/R address hierarchy.

5.3.1.6 Common Name

The Common Name attribute is an attribute of the mnemonic O/R address. This is the mnemonic O/R address attribute recommended to identify individual O/R addresses in a routing domain. MAILbus 400 MTAs use the Common Name attribute or the Numeric User Identifier for identifying individual O/R addresses.

You can use any value you choose for the Common Name attribute, subject to the restrictions of the character sets that you are allowed to use. However, it is best to use a consistent scheme, for example, given name followed by surname.

Alternatively, the Common Name attribute could identify an organizational role, such as "Marketing Manager" as well as, or instead of, the name of a person.

5.3.1.7 Personal Name

Where you plan to have connections to routing domains based on the 1984 MHS standards, you will need to register a Personal Name as well as a Common Name attribute for some users. The Personal Name attribute is required for those users in your routing domain who are exchanging mail with users of 1984 systems. This ensures that the users of the 1984 system can successfully reply to messages.

You do not need to assign a Personal Name attribute to individuals in the routing domain who will not be sending messages to 1984 systems, or if the 1984 system supports the Domain Defined attribute of type Common. *HP MAILbus 400 MTA Tuning and Problem Solving* describes the use of the Personal Name attribute and the Domain Defined attributes when downgrading a message such that it can be exchanged between 1984 and 1992 systems.

5.3.1.8 Domain Defined

Domain Defined attributes (DDAs) are intended for addressing non-X.400 messaging systems. By using DDAs these systems can continue using their addressing conventions and interwork with X.400 messaging systems.

You cannot use a DDA to name an individual in a MAILbus 400 MTA routing domain. However, a MAILbus 400 MTA can transfer messages with O/R addresses that include DDAs.

5.3.1.9 Numeric User Identifier

This attribute is used to identify users numerically relative to an ADMD or PRMD. Therefore, you can only use this attribute if your organization spans an X.400 management domain.

This attribute can be used as an alternative to the Common Name attribute, but is not as friendly.

5.4 Allocating O/R Address Attributes to Individuals in Your Routing Domain

The following is a summary of what you need to do to allocate values to the O/R address attributes that you plan to include in O/R addresses for the individuals in your routing domain. You then enter these O/R address attributes in the directory and assign routing information to them.

1. Obtain values for the mandatory attributes: Country Name and Administration Domain Name. A list of country codes is given in Appendix D. Complete one or more of the following for an Administration Domain Name:
   - Contact your ADMD to obtain an Administration Domain Name.
   - If you never intend to have connections to an ADMD, you need not contact a registration authority for an Administration Domain Name. You can use 0 (zero) as its value instead.
   - If your ADMD has an agreement with other ADMDs to relay messages from any PRMD, use a single space, " ".
2. Decide which of the conditional attributes you want to use (see Table 5–1).

   The ACME Shoe Corporation might choose the following as its conditional attribute values:
   - ACME as the value of the Private Domain Name.
   - ACME as the value of the Organization Name attribute.
   - WELL and AUCK as the value of each of the Organizational Unit Name attributes subordinate to ACME.

   Chapter 7 provides an example of how to enter the O/R address attributes that have been assigned by the ACME Shoe Corporation in the directory.
3. Identify a policy for naming individuals in your routing domain on the basis of the Common Name attribute (or Numeric User Identifier if you are using the numeric O/R address form). You can use the internal telephone directory, or the master personnel file, which hold the names of individuals

in the organization. If your organization is large, you can delegate the task of entering the O/R addresses for individuals. For example, this could be done on an MTA and User Agent basis, that is, one person could be responsible for maintaining the directory entries for individuals that connect to a particular MTA.

Table 5–2 shows an example of the attributes and associated values for the mnemonic O/R address that is being assigned to Sally Payne.

Table 5–2 O/R Address for Sally Payne

Attribute Label	Value
C	NZ
A	NZ-PTT
P	ACME
O	ACME
OU1	WELL
CN	Sally Payne

For the MAILbus 400 MTA this O/R address is expressed as follows:

```
C=NZ; A=NZ-PTT; P=ACME; O=ACME; OU1=WELL; CN=Sally Payne
```

Chapter 7 provides examples of how to create O/R address entries using the ACME Shoe Corporation as an example. Part III describes how to set up your routing domain, and when to enter the O/R address attribute values that you have planned in the directory.

5.4.1 Resolving Naming Ambiguities

If you have two or more people with the same name within the same group, you can distinguish between them by using the middle initials, if these initials are different.

For example, a new employee, John Jones, joins the ACME Shoe Corporation in Auckland. However, there is already a John Jones in Auckland whose O/R address is:

```
C=NZ; A=NZ-PTT; P=ACME; O=ACME; OU1=AUCK; CN=John Jones
```

If ACME follows its naming scheme, it has to allocate exactly the same O/R address for the new John Jones. This cannot be done, because an O/R address must be unambiguous. Therefore, ACME needs to find a way of differentiating between the two. This can be done by identifying the new John Jones using

his middle initial, if he has one. In this case, the new John Jones could be identified as John P. Jones.

```
C=NZ; A=NZ-PTT; P=ACME; O=ACME; OU1=AUCK; CN=John P. Jones
```

The original John Jones can continue to be known simply as John Jones.

5.5 Representing O/R Addresses for Individuals in Other Routing Domains

If you intend to have connections to other routing domains, you need to plan the O/R addresses that you want to store in the directory to represent the individuals, or groups of individuals, in these routing domains. You can then add routing information to these O/R addresses so MTAs in your routing domain can route messages to the individuals, or groups of individuals, in the other routing domains. In the case of other X.400 routing domains, the attribute values for these O/R addresses are provided by the organizations that operate the mail systems in the other routing domains. In the case of non-X.400 routing domains served by a Gateway, you might have to plan O/R addresses for the user community served by the Gateway. This address represents the Gateway user community within your MAILbus 400 MTA routing domain. Refer to the Gateway documentation for details of what needs to be planned for the user communities served by the Gateway.

5.5.1 Complete or Partial O/R Addresses

You need to decide whether you want to represent complete or partial O/R addresses and add the routing information to these O/R addresses accordingly. A complete O/R address uniquely identifies an individual within a user community. A partial O/R address is a series of leading attributes of an O/R address.

Where possible, represent groups of individuals in other routing domains using partial O/R addresses. However, if your routing domain connects to another routing domain through a Gateway, you need to find out whether the Gateway uses the HP Enterprise Directory Service to translate mail addresses. Whether or not the Gateway translates addresses can affect how you represent O/R addresses. In addition, some Gateways need to own an O/R address in order to provide routing information, including content information for the user community that the Gateway serves. Contact the person responsible for managing the Gateway to determine whether the Gateway is set up to use the HP Enterprise Directory Service to translate mail addresses.

In cases where there is no single partial O/R address that represents a complete user community, you can subdivide the user community and assign one or more partial O/R addresses to represent the subdivisions. You can also assign complete O/R addresses to some individuals in the other routing domain in combination with partial O/R addresses. You can then include routing information in the partial and complete O/R address entries as appropriate. Section 5.7 and Section 5.8 describe the routing information that is applicable to complete and partial O/R addresses respectively. The documentation provided with the Gateway will explain in detail how the particular Gateway translates addresses and how to plan the translations.

If the routing domain to which you plan to connect is based on the postal or telematic services, refer to Appendix A. This appendix describes the O/R address attributes for the postal and terminal O/R addresses that you store in the directory for these user communities.

5.6 Alias O/R Addresses

You can provide **alias** values for any of the O/R address attributes. There are several instances in which you might want to assign an alias O/R address:

- Where an attribute can have more than one value, for example, Printable and Teletex values, or multiple values for O/R address attributes.

  As an example, routing domains based on the 1984 MHS Standards support only the Printable string character set. If you want to interwork with a system based on the 1984 MHS Standards, you provide alias entries in the directory for O/R address attributes that have both a Teletex and Printable representation. In this way, individuals within the routing domain can address each other using the teletex values, but must include the printable representation when sending messages to individuals in routing domains based on the 1984 MHS standards. However, if you have a policy for using printable string values for O/R address attributes used within your routing domain, you do not need alias O/R address entries.

- Where there are alternative spellings for any of the attributes of the O/R address. For example, William Laurence might choose to include the following alternative spellings for his name:

  William Laurence
  Bill Laurence
  William Lawrence
  Bill Lawrence

- Where your routing domain has more than one GDI.

If your routing domain has more than one GDI, you create entries for all the attributes of the GDI. You then create an alias entry that directs all the GDIs to a common point on the O/R address hierarchy for your routing domain. This allows the flexibility of specifying different routing information for the GDI attribute values, and common routing information for individuals, or groups of individuals in the routing domain.

You also need to include all GDIs as an attribute of the MTA entry in the directory.

Chapter 7 shows how you enter multiple GDIs and associated routing information in the directory using ACME as an example.

5.7 Routing Information for Complete O/R Addresses

You must provide the routing information for the O/R address attributes representing individuals in your routing domain, and in some cases, for the individuals in the routing domains to which your routing domain connects. This routing information is an attribute of the complete O/R address entry, usually the Common Name attribute, and specifies the action that an MTA takes when it routes a message addressed to the complete O/R address.

If you are only representing complete O/R addresses so that a Gateway can translate complete O/R addresses to foreign mail addresses, you do not need routing information for each complete O/R address. Place the routing information at the partial O/R address entry that represents the entire Gateway user community. If there is no single partial O/R address entry that represents this user community, include the routing information at each of the appropriate partial O/R address entries that together represent the user community.

Note that the attributes of the partial O/R addresses that make up a complete O/R address can also include routing information. The routing information that you can specify for partial O/R addresses is described in Section 5.8.

Routing information cannot be specified for alias O/R address entries. Where you have aliases, enter the routing information as an attribute of the O/R address entry that is not the alias.

The routing information that you can specify for a complete O/R address comprises:

- A routing instruction that indicates:
  - That messages are to be transferred to the MTA and delivered to the Agent that serves the O/R address.

In the case of the O/R address being served by a registered Agent, you need to identify the name of the registered Agent and the name of the MTA that serves the O/R address.

In the case of an unregistered Agent, the O/R address represents the unregistered Agent. You need to identify the name of the MTA that serves the O/R address. You may also be required to specify a password for the O/R address. Consult the person responsible for managing the Agent to find out whether a password is required.

- An MTA is to transfer messages to the routing domain that serves the O/R address.

  If the O/R address is for an individual in another routing domain, you need to specify the name of the routing domain that serves the O/R address.

  If you use this routing instruction, you need to provide routing information that indicates how an MTA in your routing domain connects to the other routing domain. The routing information that you provide for other routing domains is described in Section 5.9.

- An MTA is to non-deliver all messages for an O/R address.

  There are cases where you might want to indicate explicitly that all messages for a particular O/R address must not be delivered. For complete O/R addresses you might do this if the individual with the O/R address has left the organization.

  Where it can, the MTA sends a non-delivery report to the message originator.

- Redirection to another O/R address.

  An individual might want to be temporarily known at a different O/R address. In this case, you can specify a redirection address as routing information indicating that MTAs must redirect messages for a particular O/R address to another O/R address.

  Recipient redirection is specified on complete O/R addresses for the temporary redirection of a recipient's messages. This is the MT element of service "Redirection of Incoming Messages" as defined in the 1992 MHS Standards.

- Definitive O/R Address.

  When specified, the MTA passes the definitive O/R address to the User Agent as part of message delivery for a particular recipient. A definitive O/R address can only be used by User Agents using the XAPI interface.

You require a definitive O/R address where a user's O/R address on a message can be specified in more than one way, for example, where a user has alias O/R addresses for multiple GDIs or more than one way of spelling his or her name. In these cases, a User Agent needs a way of equating all a user's alias addresses to the user. It does this by using the definitive O/R address. The definitive O/R address is the same for all O/R addresses that represent a particular user.

You are recommended to specify a definitive O/R address for all recipients who use a registered Agent. Note that the MAILbus 400 Message Store requires all individuals who use the Message Store to have a definitive O/R address.

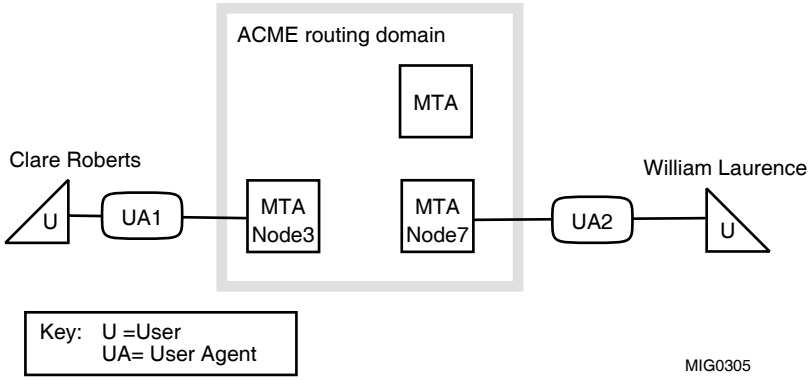
Make sure that the O/R address that your User Agent recognizes as the definitive O/R address is correct for each recipient that the Agent serves.

- Personal names.

  You might need to represent personal names in the directory for individuals in your routing domain, in addition to the common names already planned. Section 5.3.1.7 explains when you require a personal name.

The following example describes the routing information that you need to enter in the directory for two individuals in the ACME routing domain. The layout of the routing domain for this example is shown in Figure 5–3.

Figure 5–3 Individuals in Your Routing Domain



The routing information that you need in order that MTAs in your routing domain know how to deliver messages to these individuals comprises the following:

- Entries in the directory representing the complete O/R addresses for Clare Roberts and William Laurence and routing instructions at these O/R address entries indicating the name of the MTA and User Agent by which they are each served.

  The routing instruction for Clare Roberts' O/R address specifies that the O/R address is served by the MTA "MTA-NODE3" and the User Agent "UA1".

  The routing instruction for William Laurence's O/R address specifies that the O/R address is served by the MTA "MTA-NODE7" and the User Agent "UA2".

- Information about the registered User Agents UA1 and UA2, which is held locally at the MTA where each User Agent is located. This information is represented by the Agent entity, an entity of the MTA module, as described in Section 4.9.

Chapter 7 provides examples of how you enter O/R addresses and associated routing information in the directory.

5.8 Routing Information for Partial O/R Addresses

You need to identify the routing information for each of the partial O/R address entries in the directory. These could be partial O/R addresses representing groups of individuals in other routing domains, or partial O/R addresses that are also part of complete O/R addresses. *HP MAILbus 400 MTA Tuning and Problem Solving* provides an example of how the MTA uses this routing information to route messages.

Note that routing information cannot be specified for alias entries. Where you have aliases, enter the routing information as an attribute of the O/R address entry in the directory that is not an alias.

This routing information is in addition to that already planned for complete O/R addresses that represent individuals in your routing domain and in other routing domains.

The routing information that you can specify for a partial O/R address is a routing instruction that indicates:

- An MTA is to transfer messages to the routing domain that serves the O/R address.

  Where the O/R address is for a group of individuals in another routing domain, you need to specify the name of the routing domain that serves the O/R address.

  If you use this routing instruction, you need to provide routing information that indicates how an MTA in your routing domain connects to an MTA in the other routing domain. The routing information that you need to provide for the other routing domain is described in Section 5.9.

- An MTA is to deliver messages to the MTA and Agent that serve the O/R address.

  In the case of the O/R address being served by a registered Agent, you need to identify the name of the registered Agent and the name of the MTA that serves the O/R address.

- An MTA is to non-deliver all messages for an O/R address.

  There are cases where you might want to explicitly indicate that all messages for a particular O/R address are not delivered. For example, you might want to prevent the delivery of messages to a particular country, or ADMD. You can also specify this routing instruction to prevent the MTA looking for alternative routing information on less specific O/R address entries.

  Where it can, the MTA sends a non-delivery report to the message originator.

- An MTA is to redirect messages to another O/R address.

  Redirection, in this case management domain (MD) Redirection, is for the temporary redirection of messages. Typically, MD Redirection is used for redirecting incorrectly addressed messages to a known destination and is used only with partial O/R addresses. For example, you might choose to redirect all messages with the partial O/R address representing the organization ACME to a secretary of the ACME Shoe Corporation. In this way incorrectly addressed messages within the ACME organization are dealt with at a central point.

  This is the MT element of service “Alternate Recipient Assignment” as defined in the 1992 MHS Standards.

Figure 5–4 Individuals in Other X.400 Routing Domains

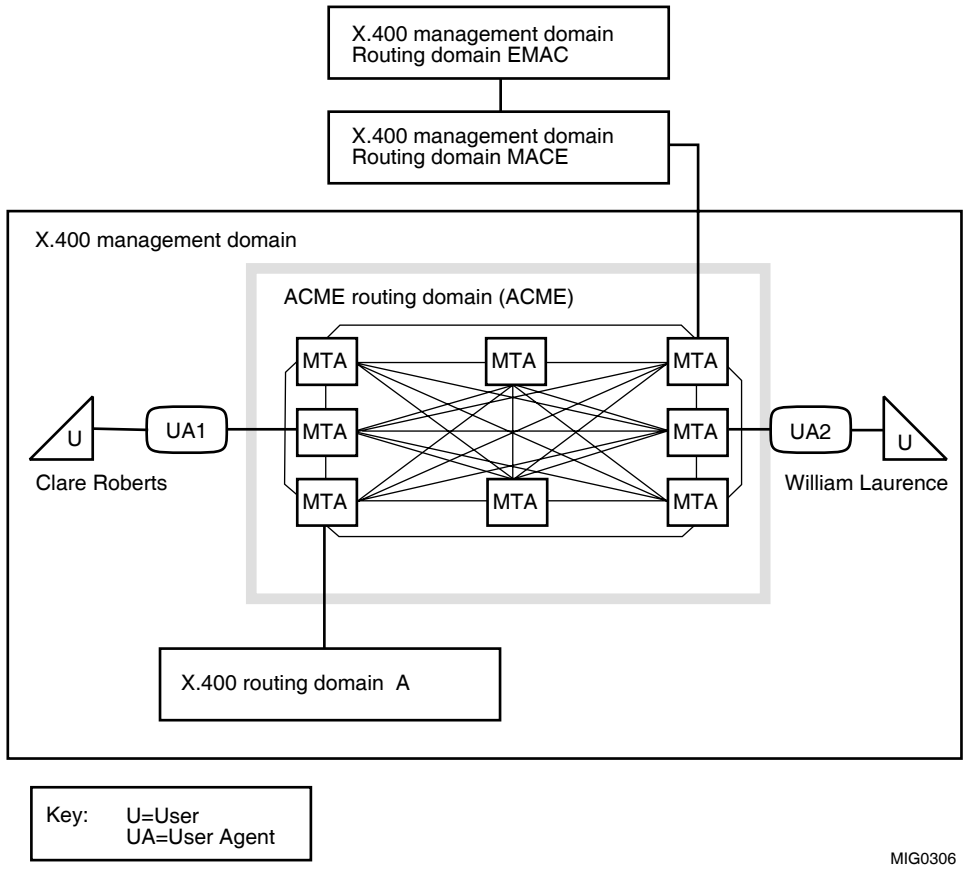


Figure 5–4 shows individuals and groups of individuals both within your routing domain and in other routing domains. The routing information that you need to define so order that MTAs in your routing domain know how to deliver messages to these individuals comprises:

- Entries in the directory representing the partial O/R addresses for the user communities served by the other routing domains.

  Routing instructions for each partial O/R address entry that identifies the routing domain that serves the O/R address.

- Entries in the directory that represent the routing domains to which your routing domain connects, either directly or indirectly: “A”, “MACE”, and “EMAC”.

  Routing instructions for these routing domains as described in Section 5.9.

  In addition, for routing domains “MACE” and “EMAC”, you need to indicate that the routing domains are part of other X.400 management domains.

- Peer MTA entities, which are entities of the MTA module, held locally at the boundary MTAs. These Peer MTA entities provide details of how the boundary MTAs communicate with the routing domains “A” and “MACE”.

Chapter 7 provides examples of how you enter partial O/R addresses and associated routing information in the directory.

5.9 Routing Information for Other Routing Domains

This section describes the routing information that you can specify for any other routing domains to which your routing domain will connect. In Section 5.7 and Section 5.8 you have planned the routing information for each of the partial O/R addresses representing groups of individuals in other routing domains. Some of this information identifies the name of the routing domains that serve the O/R addresses. You now need to consider how to connect to these routing domains from your routing domain. This routing information is represented as attributes of domain entries in the directory.

A domain entry contains routing information that specifies the action that an MTA is to take when routing messages to recipients in that routing domain. This action can be one of:

- Transfer messages for the routing domain to a boundary MTA.

  Where MTAs in your routing domain connect directly to MTAs in another X.400 routing domain, you need to identify the name of the boundary MTA in your routing domain. There is additional routing information that is held at the boundary MTA to provide the communications information between the boundary MTA and the peer MTA in the other routing domain. This information is held in the Peer MTA entity, an entity of the MTA module, and is described in Section 4.8. As an example, the domain entry for the routing domain “A” indicates that routing domain “A” is served by “MTA-NODE4”, as shown in Figure 4–5.

- Transfer messages for the routing domain to a Gateway.

Where MTAs in your routing domain communicate with systems in another routing domain through a Gateway, as shown in Figure 4–6, you need to identify the name of the Gateway. You also need to identify the MTA of which the Gateway is an Agent. With reference to Figure 4–6 the routing domain named “SMTP” is served by the MTA “MTA-NODE8” and an Agent named “SMTP”.

There is additional routing information that is held at the MTA that knows the Gateway. This information is held in the Agent entity, an entity of the MTA module, and is described in Section 4.9.

- Transfer messages for this routing domain through another routing domain.

  Where MTAs in your routing domain connect indirectly to MTAs in another routing domain, identify the name of the intermediate routing domain that connects to the other routing domain. As an example, the entry for the routing domain “EMAC” identifies “MACE” as the intermediate routing domain.

Chapter 7 provides examples of how you enter information in the directory describing other routing domains and how these routing domains are served from your routing domain.

5.10 Routing Information for Different Forms of O/R Address

There are four forms of O/R address: Mnemonic, Numeric, Postal and Terminal. O/R address forms have some common terms that the MTA uses for routing, namely the Country Name, Administration Management Domain Name, and Private Management Domain Name attributes. If you want an MTA to route according to the form of O/R address for any of these attribute values, when there are no other O/R address attribute values in the directory, you need to use routing instructions that are specific to the form of O/R address.

If you want the MTA to route messages to the same destination whatever the form of O/R address, you do not need to use specific forms of routing instruction.

This allows you to specify a routing instruction that is specific to one O/R address form in addition to another routing instruction that is used for all other forms of O/R address. For example, you can add a routing instruction for a telematic O/R address, and a non-specific routing instruction for all other forms of O/R address.

An example of using routing instructions for different forms of O/R address is given in *HP MAILbus 400 MTA Tuning and Problem Solving*.

5.11 Content Information

In addition to the routing information, you can also add Content Information at any point in your O/R address hierarchy. Content Information is used by MAILbus 400 MTAs to determine whether or not a particular message size, message content type or bodypart type is acceptable to a recipient's User Agent. As a result an MTA can decide whether to deliver, convert, or not deliver a message. As an example, you could have a group of users who are part of a different organization that can only receive messages of a certain content type or bodypart type. In this case, you could place the Content Information in the partial O/R address entry that represents the organization.

You are not required to add Content Information in order to have an operational MTS. However, if you want to specify that users within your routing domain receive messages of a particular content type or bodypart type, refer to Part II of *HP MAILbus 400 MTA Tuning and Problem Solving*, which describes the details of specifying Content Information for O/R addresses.

You can specify Content Information attributes that are specific to each form of O/R address. Use these attributes where it is not possible to distinguish the form of O/R address from the attributes that are represented in the directory. These are the Country Name, Administration Management Domain Name, and Private Management Domain Name attributes.

5.12 Distribution Lists

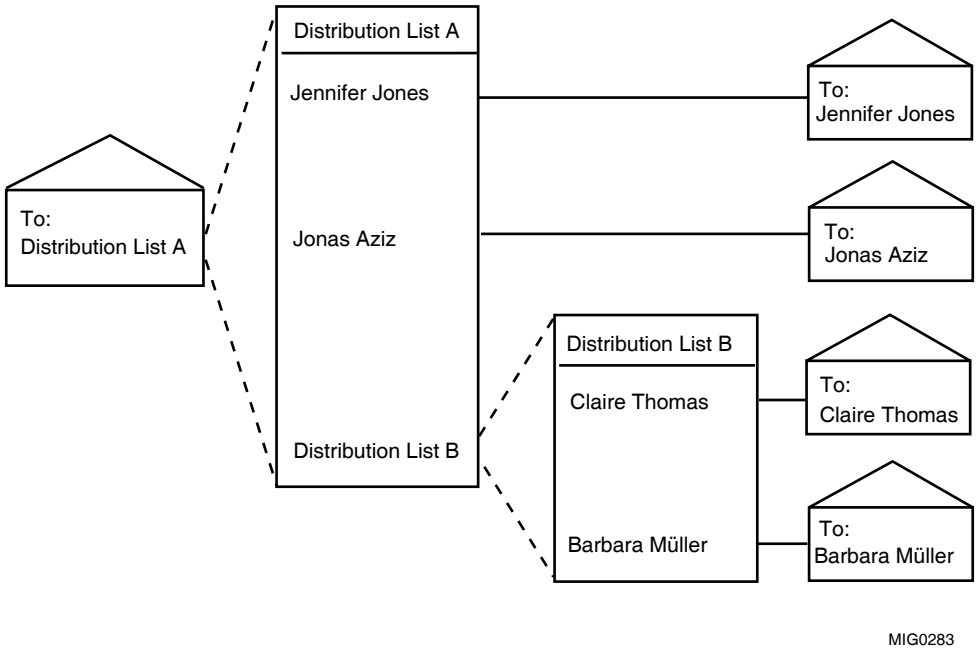
You can group the O/R names of the intended recipients of a message together to form a **distribution list**. A distribution list is represented by an O/R name and can contain the O/R names of individual users as well as the O/R names of other distribution lists (known as nested distribution lists).

O/R names of distribution lists are used in the same way as the O/R names of users. To send a message to all recipients in the distribution list the originator simply needs to specify the O/R name of the distribution list. When an MTA processes the message, it expands the distribution list to send the message to all users in the distribution list, whether specified individually or indirectly through nested distribution lists.

The MAILbus 400 MTA only processes O/R names that contain O/R addresses. This means that the MAILbus 400 MTA does not expand a distribution list if its O/R name only contains a Directory name. The MAILbus 400 MTA also cannot deliver messages to those recipients in a distribution list whose O/R names contain only Directory names.

Figure 5–5 illustrates the principle of distribution list expansion. For illustrative purposes this example uses names rather than O/R names.

Figure 5–5 Distribution List Expansion



Refer to the *MTS Module Online Help* and the planning summary based on the ACME Shoe Corporation (Chapter 7) for details of how to represent distribution lists in the directory. An example based on the ACME Shoe Corporation is given in Appendix B.

5.13 Permission to Send Mail to Another X.400 Management Domain

You can prevent an individual in your routing domain from sending messages to individuals located within a different X.400 management (CCITT) domain. You do this by adding permission information to the individual's O/R address (in your routing domain) in the directory.

You cannot prevent a user within your routing domain from receiving messages from individuals in other X.400 management domains and you cannot prevent individuals in other routing domains from sending messages to recipients within your routing domain.

However, you can prevent individuals in other X.400 management domains from sending messages across your routing domain to a different X.400 management domain. You do this by adding permission information to the O/R addresses of the individuals (in the other X.400 management domain) in the directory.

Permission information that is set for a partial O/R address entry applies to all individuals represented by the partial O/R address. It also applies to any more complete O/R address entries that are subordinate to the partial O/R address entry, where these O/R addresses do not have any permission information set.

Unless you explicitly indicate otherwise, individuals and applications have permission to exchange messages with users of any other X.400 management domain to which your routing domain connects. Chapter 7 provides an example of how to add permission information to an O/R address.

5.14 Warning Text on Messages Arriving from Other X.400 Management Domains

You can specify warning text on messages that arrive from other X.400 management domains. Specifying warning text is described in *HP MAILbus 400 MTA Tuning and Problem Solving*.

5.15 Recovering Messages from an MTA/recover_msg

If an MTA is shut down, you can use message recovery to enable another MTA in the same routing domain to take responsibility for the messages that were waiting to be processed when the MTA was shut down. When an MTA takes responsibility for a peer MTA's messages, it attempts to process those messages as the peer MTA would have done.

HP MAILbus 400 MTA Tuning and Problem Solving describes in detail how to use message recovery.

5.16 What to Do Next

Read Chapter 6 and Chapter 7, which describe planning the use of the HP Enterprise Directory Service and examples of how routing information is stored.

There are some additional features of the MTA that are not covered in planning. You can plan and implement these additional features at any time after you have set up the MTAs in your routing domain and stored all the O/R addresses and routing information in the directory. These are:

- Message conversion and downgrade as a result of specifying Content Information.
- **Accounting**

  You can use Accounting to record particular items of information about the messages that enter and leave your routing domain.
- **Archiving**

  You can use Archiving to store exact copies of messages on disk, for example, if you need to keep such copies for legal reasons.
- **Message History logging**

  You can use Message History logging to keep information at an MTA about the messages (excluding reports and probes) that it has processed. This information enables you to trace messages in your routing domain.

Part II of *HP MAILbus 400 MTA Tuning and Problem Solving* describes these features and how to implement them.

6

Planning the Use of the HP Enterprise Directory Service

This chapter describes what needs to be considered and set up for MAILbus 400 MTAs to use the HP Enterprise Directory Service.

If you have already set up the directory as a result of reading *MAILbus 400 Getting Started*, you need only read this chapter if you intend to set up more than one DSA in your routing domain. If you do not intend to set up more than one DSA, continue at Chapter 7.

There might be someone responsible for managing the HP Enterprise Directory Service, in which case you need to plan the use of the directory in cooperation with this person. However, you might be managing the directory exclusively for the MAILbus 400 MTA, in which case you need to read the HP Enterprise Directory Service documentation in conjunction with this chapter.

6.1 The MTA and the HP Enterprise Directory Service

The MAILbus 400 MTA uses the HP Enterprise Directory Service to store routing information that needs to be shared between MAILbus 400 MTAs in your routing domain. You enter O/R addresses and routing information in the HP Enterprise Directory Service only once, at one chosen DSA. This information is shared by all your MTAs, which ensures that the routing actions taken by your MTAs are consistent, and avoids the need for you to plan message routes for each individual MTA.

You can choose to use the HP Enterprise Directory Service in a centralized way, where all your MTAs use the same directory system agent (DSA). In this case there is a single copy, the master copy, of the routing information held at this DSA.

Alternatively, you can choose to have more than one copy of the routing information, with each copy held at a particular DSA. This is known as **replication**. In this case, there is one master copy and a number of shadow copies of the routing information.

If you choose to replicate your routing information such that some MTAs obtain their routing information from DSAs that hold shadow copies, you need to plan which MTA will be contacting which DSA for its routing information, and where the individual MTAs will be installed. This needs to be done taking into account other directory applications and which DSAs they use.

Note

When you install and set up your MTAs for the first time, do not set up the MTAs to contact a DSA that holds a shadow copy of the MTA's routing information. Set up each MTA to contact the DSA that holds the master copy of the MTA's routing information. *Part III, Setup* describes how to set up each MTA in your routing domain.

You might choose to replicate the MTA routing information, if you have a large organization and want your applications in several locations to benefit from having access to the same information at all locations.

6.2 Regulating Access to the Directory

The MAILbus 400 MTA might not be the only application that is using the HP Enterprise Directory Service. Other applications might also use the HP Enterprise Directory Service to store their information.

To ensure that the MTA's directory information is safe from unauthorized access or modification attempts, the MTA imposes access restrictions on its directory information. This ensures that only authorized users, or applications such as the MTA itself, can access or modify the MTA's information in the directory.

6.3 HP Enterprise Directory Service Setup Required

The following list describes what needs to be set up in the HP Enterprise Directory Service in order for you to use the directory to store the routing information for your routing domain.

- A Naming Context entity for the MTAs' routing information.

  A Naming Context entity, which is an entity of the DSA management module, must be created for your routing domain entry. The routing domain entry is the entry beneath which all the routing information for your routing domain is stored. This is described in Section 4.1.1. You need to know at which DSA this naming context is created, as it will be the master DSA for the routing information for your routing domain.

You must create only one naming context for the routing domain. Do not divide the routing information for a routing domain into more than one naming context. This is because the entire naming context for the routing domain must be available to each MTA that is part of the routing domain.

You cannot create any entries in the directory for the routing domain until you have created this Naming Context entity in the HP Enterprise Directory Service. Use the distinguished name chosen in Chapter 4 for the routing domain as the identifier for the Naming Context entity. This is also the identifier of the MTS entity, an entity of the MTS module, which is described in more detail in Chapter 7.

Part III, Setup provides an example of how to create a Naming Context entity.

- A Subordinate Reference entity.

  If there is a naming context immediately superior to the naming context for the routing domain, make sure that there is a Subordinate Reference entity created for the routing domain entry. This Subordinate Reference entity is created on the node where the master DSA for the superior naming context is set up.

  If a subordinate reference is required, it must be created before the naming context for the routing domain.

  Note that if you create your routing domain entry at root, you will not need a Subordinate Reference entity.

6.4 Ensuring Routing Information is Available to MTAs

Ensuring that the MTAs have access to routing information in the directory can be done in a number of ways. You will need to consult the person responsible for managing the directory, or the HP Enterprise Directory Service documentation, to determine the best method to use. You will need to take into account the following:

- Access control and DSAs.

  Ask the person responsible for managing the directory to provide you with write-access to the directory. If you do not have write-access, you will not be able to create the MTA's routing information. However, if you have installed the directory solely for the MTA, you will not need authority as the directory is installed with write-access for all users.

- Applications sharing DSAs.

  Find out whether there are other directory applications installed on the nodes where you want to install your MAILbus 400 MTAs. If there are, you need to make sure that the DSAs that these applications use hold copies of your routing information.

  If there are no directory applications installed on the node, you can decide to which DSA you want the MTA on the node to connect, and you can make sure that the DSA has a copy of the routing information for your routing domain.

- Replicating the MTAs' routing information.

  If you intend to replicate the routing information for your routing domain, use the following guidelines:

  – Replicate the naming context for your routing domain to each shadow DSA to which any MTA connects.

  – Make arrangements for the DSA Update command to be issued at each shadow DSA that holds a copy of the routing information for your routing domain whenever you modify routing information in the directory. Be aware that unless all the MTAs in the routing domain have the same routing information available to them, inconsistencies in the messaging service might be seen.

The routing information, or a copy, must be available to the DSA to which an MTA connects. You are advised not to configure a DSA used by an MTA to use Referrals. You must use Replication if you want to distribute your routing information across multiple DSAs.

7

Summary of Managing Routing Information

This chapter summarizes, in the form of a worked example, the important routing concepts that you need to understand before you set up your routing domain. In particular, you need to be aware that although most routing information is held in the directory and shared between MTAs, some routing information is held locally at individual MTAs. Routing information held in the directory is used by the MTA that is to deliver or transfer the message, and is used by MTAs to find out the destination of a message with a particular O/R address. Routing information held locally at individual MTAs provides information about how to connect to specific Agents or Peer MTAs.

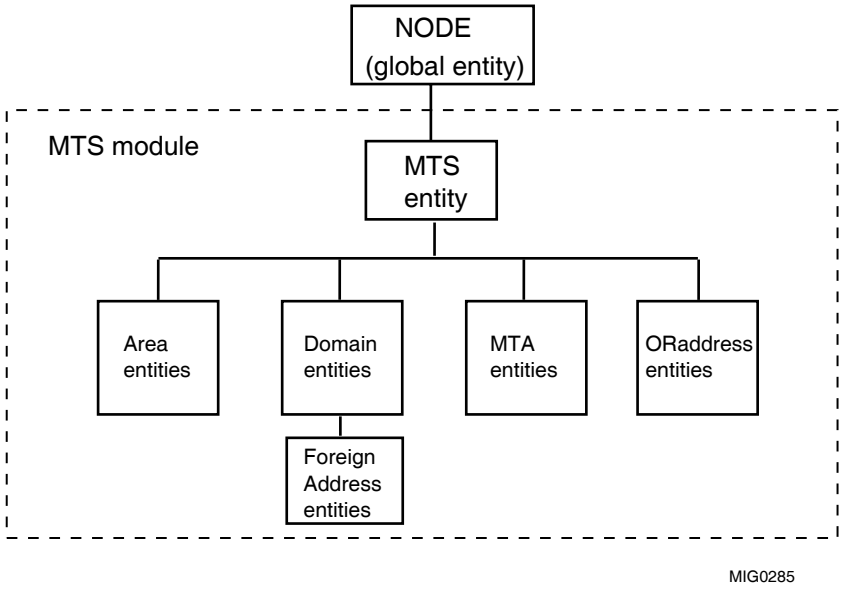
The shared routing information is entered in the directory once. Figure 7–1 shows the structure of the MTS module, which is one tool that can be used to enter routing information in the directory. You can also use HP X.500 Information Manager to enter the MTA's routing information in the directory. However, all the examples in this guide show the MTS module as the tool for entering routing information.

The information held locally at MTAs is entered as part of the MTA startup script, using the MTA module, each time an MTA is created.

7.1 The MTS Module

In terms of EMA management, the MTS entity, an entity of the MTS module, is a subentity of the global entity NODE. However, the information represented by the MTS entity is not specific to the node, but is shared in the directory by all the MTAs, and some Agents, that specify the MTS entity name when they are created.

Figure 7–1 MTS Module and Entities



You might be planning to have more than one MAILbus 400 MTA routing domain in your network, in which case, you will have more than one MTS entity. When you set up an MTA, you specify the distinguished name for the routing domain of which the MTA is a part, and the MTA's password. This ensures that MTAs in the same routing domain in your network are all using the correct routing information.

7.1.1 Replication

If you choose to replicate your routing information in the HP Enterprise Directory Service, be aware that there might (temporarily) be inconsistencies in the routing information that is available to individual MTAs. For example, if you make changes to routing information at a particular DSA, these changes are not made available to all the shadow DSAs in the routing domain until the shadow DSAs have been updated. The frequency of these updates is determined by the person responsible for managing the directory.

When entering routing information using the MTS entity, use an MTS entity that has access to the master DSA. If you do this, you can be sure that you can confirm that you have entered the information correctly. If you use an MTS entity that has access to a shadow DSA, the shadow DSA passes requests for modifications to the master DSA. This means that if you modify an entry in the

directory at a shadow DSA and then show the same entry, you will not see the modification. You only see the modification after the shadow DSA is updated.

7.1.2 Access Control

Access to the routing information represented by a particular MTS entity is controlled, so only users who know the password for the routing information can modify routing information. Each MTS entity that you create requires a unique password.

When you create a routing domain entry using the MTS entity in the directory, you must provide a password as an attribute of the routing domain entry. You will need additional authorization to create the routing domain entry if you do not have write-access to the point in the directory hierarchy where you want to create the routing domain entry.

If you want to manage routing information that you have entered at one node from other nodes, you must first provide the necessary authorization, by quoting the password.

If you are directing your MTS entity commands to another node, the node specified in the command, and not the node where you are issuing the command, requires the authorization. The following is an example of the command to use if the MTS entity on the node where you are directing your MTS entity commands is not authorized:

```
AUTHORIZE NODE MTA-NODE1 MTS "/MTS=ACME" -
PASSWORD "unforgettable-mts-password"
```

Applications that read routing information from the directory might also require a password. In the case of the MTA, a password is always required and is supplied when the MTA is set up. In the case of other applications such as User Agents or Gateways, a password might be required for the O/R address or Domain entry that represents the Agent. The documentation supplied with the application will describe how the particular application controls its access to the directory.

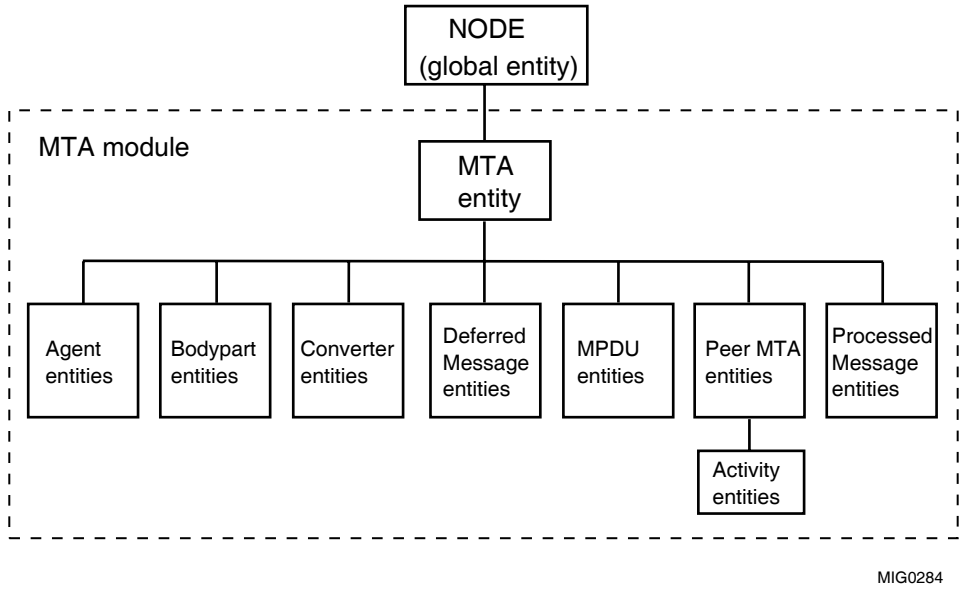
Part III, Setup describes how to set up your routing domain, including setting up access control. *MTS Module Online Help* describes the Password attribute and the Authorize command for the MTS entity in more detail.

7.2 Routing Information Held Locally at an MTA

The routing information required locally at individual MTAs is held by two entities of the MTA module: the Agent entity and the Peer MTA entity (see Figure 7–2). These entities hold information about the Agents that an MTA serves and about peer MTAs in other routing domains to which a boundary MTA in your routing domain connects. This information is not held in the directory.

The remaining entities of the MTA module are used to manage the MTA and are described in *HP MAILbus 400 MTA Tuning and Problem Solving*.

Figure 7–2 MTA Module and Entities



The routing information that you need to set up at each MTA in your routing domain comprises:

- Routing information held locally at individual MTAs where Agents are located. This information is held in Agent entities of the MTA module.

- Routing information held locally at individual MTAs that are boundary MTAs to other X.400 routing domains. This information is held in Peer MTA entities of the MTA module.

The complete set of attributes for the Agent and Peer MTA entities are given in *HP MAILbus 400 MTA Tuning and Problem Solving*.

The MTA startup script includes examples of the commands required to create Agent and Peer MTA entities. The syntax for these commands is described in the *MTA Module Online Help*.

7.3 Routing Information Held in the Directory

The routing information that you need to store in the directory for your routing domain comprises:

- O/R addresses and routing information for the individuals within your routing domain and the individuals or groups of individuals that you want to communicate with in other routing domains.

  The most important routing information is the Routing Instruction that indicates how each of the O/R addresses is served, that is, whether the O/R address is served by an MTA and User Agent, or whether the O/R address is served by another routing domain.

  This information is held in the directory in ORaddress entities of the MTS module.

- Routing information that indicates how a particular routing domain is served, that is, whether it is served by an MTA and Gateway, a boundary MTA or indirectly through another routing domain.

  This information is held in the directory in Domain entities of the MTS module.

Appendix C lists the complete set of attributes of the MTS entity that can be specified.

Note

Do not include the commands to create directory entries using the MTS module in the MTA startup script.

There is an example population script that contains the commands required to create some of the directory entries for the ACME routing domain (see Appendix B). The example population script is also one of the files installed with the MTA. Refer to the list of files in *HP MAILbus 400 MTA Tuning and*

Problem Solving for details of the location of the example script. Refer to the *MTS Module Online Help* for details of the syntax of the commands used to create all directory entries.

The remainder of this chapter provides an example of some of the routing information that represents the ACME Shoe Corporation. The example does not specify the complete set of routing information for ACME but highlights some of the important points mentioned in *Part II, Planning*.

7.4 O/R Address Information

This section describes how the O/R addresses and associated routing information that ACME has specified for its routing domain are stored in the directory.

Figure 7–3 is an example of how O/R addresses for the ACME Shoe Corporation are represented in the directory. The example includes two complete O/R addresses for Clare Roberts and William Laurence located within ACME's routing domain, and a partial O/R address to the routing domain EMAC that represents the community of users in the EMAC routing domain.

Note that you cannot create ORaddress entities until you have set up the MTA's routing domain entry as described in Part III.

The numbers in Figure 7–3 are explained in the following list:

1 This entry is for the routing domain ACME. The entry is identified by the name of the routing domain, the MTS entity identifier.

An attribute of this entry is the list of GDIs for the ACME routing domain. These are: C=NZ;A=NZ-PTT;P=ACME and C=AU;A=AU-PTT;P=ACME. These GDIs are also O/R address entries as shown in Figure 7–3.

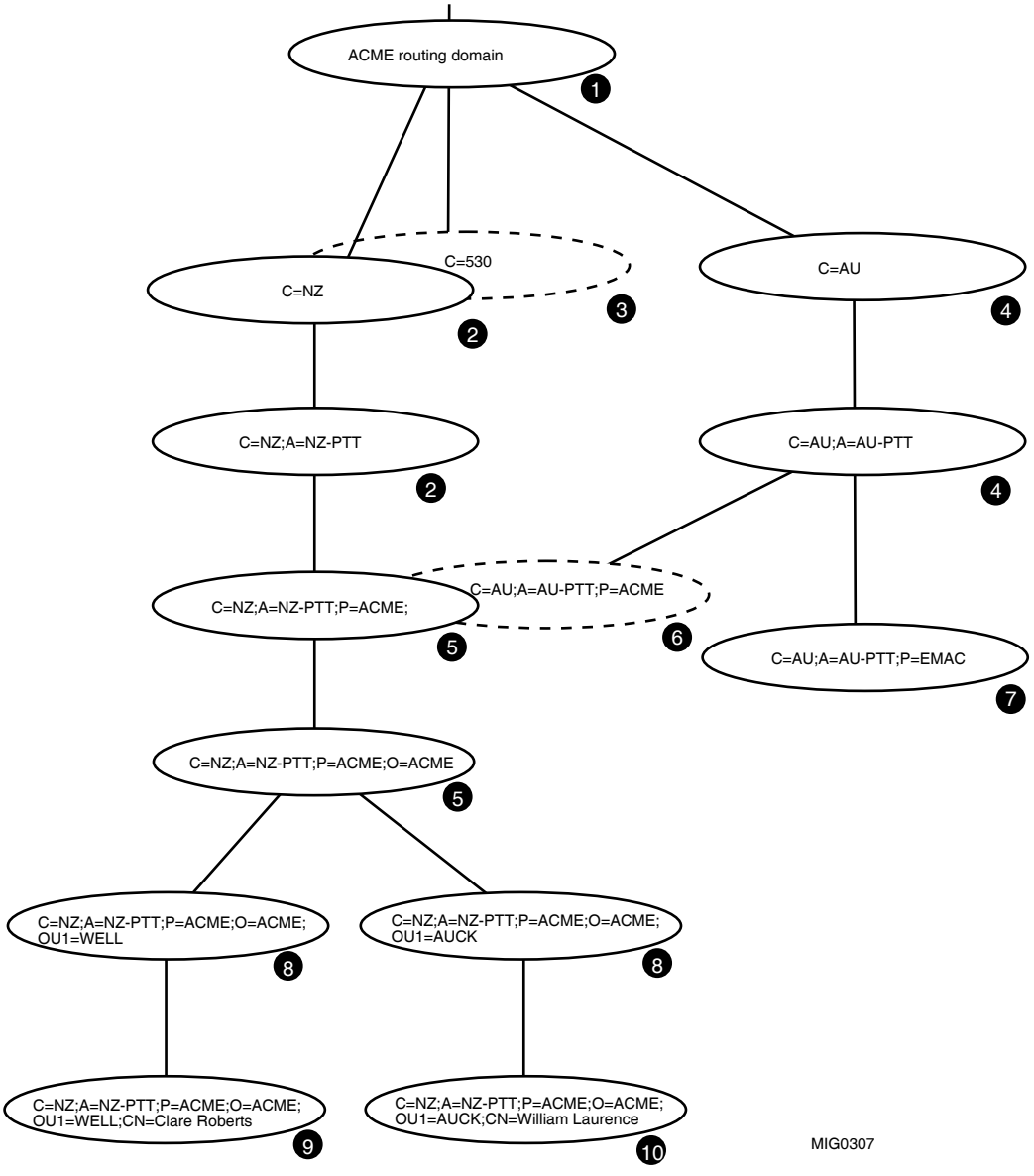
You create the routing domain entry for ACME in the directory as follows:

```
CREATE MTS "/MTS=ACME" PASSWORD "unforgettable-mts-password"
SET MTS "/MTS=ACME" GLOBAL DOMAIN IDENTIFIERS -
    {"C=NZ;A=NZ-PTT;P=ACME","C=AU;A=AU-PTT;P=ACME"}
```

2 These are the entries for the O/R addresses: C=NZ and C=NZ;A=NZ-PTT.

Routing information for these O/R address entries specifies that where no more detailed information about a recipient whose O/R address includes "C=NZ" or "C=NZ;A=NZ-PTT" is available in the directory, the MTA will route the message to the New Zealand PTT. Figure 7–4 shows the New Zealand PTT represented as the Domain entity NZ-PTT.

Figure 7–3 Directory Entries for O/R Addresses



```
CREATE MTS "/MTS=ACME" ORADDRESS  "C=NZ" -
       ROUTING INSTRUCTION [ACTION=TRANSFER TO DOMAIN, -
       SERVER DOMAIN="NZ-PTT"]

CREATE MTS "/MTS=ACME" ORADDRESS  "C=NZ;A=NZ-PTT" -
       ROUTING INSTRUCTION [ACTION=TRANSFER TO DOMAIN, -
       SERVER DOMAIN="NZ-PTT"]
```

3 This is an alias entry for the O/R address C=NZ. It provides an entry for this O/R address specifying a numeric Country Name in addition to the alphabetic Country Name. The O/R address C=530 is represented by an ORaddress entity with the O/R address as its identifier.

```
CREATE MTS "/MTS=ACME" ORADDRESS  "C=530" -
    TYPE ALIAS, ALIAS TARGET "C=NZ"
```

4 These are entries for the O/R addresses: C=AU and C=AU;A=AU-PTT.

Routing information for these O/R address entries specifies that where no more detailed information about a recipient whose O/R address includes "C=AU" or "C=AU;A=AU-PTT" is available in the directory, the MTA will route the message to the Australian PTT. Figure 7–4 shows the Australian PTT represented as the Domain entity AU-PTT.

```
CREATE MTS "/MTS=ACME" ORADDRESS  "C=AU" -
       ROUTING INSTRUCTION [ACTION=TRANSFER TO DOMAIN, -
       SERVER DOMAIN="AU-PTT"]

CREATE MTS "/MTS=ACME" ORADDRESS  "C=AU;A=AU-PTT" -
       ROUTING INSTRUCTION [ACTION=TRANSFER TO DOMAIN, -
       SERVER DOMAIN="AU-PTT"]
```

5 These are entries for the O/R addresses: C=NZ;A=NZ-PTT;P=ACME and C=NZ;A=NZ-PTT;P=ACME;O=ACME.

Routing information for the O/R address C=NZ;A=NZ-PTT;P=ACME indicates a redirection O/R address for a Secretary of the ACME Shoe Corporation. In this way, messages that are addressed to O/R addresses that include "C=NZ;A=NZ-PTT;P=ACME" and that have no more detailed O/R address attributes stored in the directory, can be delivered within the PRMD to the Secretary.

```
CREATE MTS "/MTS=ACME" ORADDRESS "C=NZ;A=NZ-PTT;P=ACME" -
       ROUTING INSTRUCTION [ACTION=MD REDIRECT, -
       REDIRECTION="C=NZ;A=NZ-PTT;P=ACME;CN=ACME Secretary"]
```

Note that you also need to create an O/R address entry for the ACME Secretary as one of the employees of the ACME Shoe Corporation.

In this example, the entry for the O/R address C=NZ;A=NZ-PTT;P=ACME;O=ACME does not have an attribute representing a Routing Instruction. However, the O/R address C=NZ;A=NZ-PTT;P=ACME;O=ACME needs to be created as part of the O/R address hierarchy.

```
CREATE MTS "/MTS=ACME" ORADDRESS -
       "C=NZ;A=NZ-PTT;P=ACME;O=ACME"
```

6 This is an alias entry for the O/R address C=NZ;A=NZ-PTT;P=ACME, one of ACME's GDIs. The O/R address C=AU;A=AU-PTT;P=ACME is created as an alias to the entry C=NZ;A=NZ-PTT;P=ACME as follows:

```
CREATE MTS "/MTS=ACME" ORADDRESS  "C=AU;A=AU-PTT;P=ACME" -
    TYPE ALIAS, ALIAS TARGET "C=NZ;A=NZ-PTT;P=ACME"
```

You create alias entries for GDIs where your routing domain is multi-homed. In this case, for the ACME Shoe Corporation, the MTA can route to recipient O/R addresses that include C=AU;A=AU-PTT;P=ACME or C=NZ;A=NZ-PTT;P=ACME.

7 This is the entry for the O/R address C=AU;A=AU-PTT;P=EMAC.

Routing information for this entry indicates that the community of users, whose O/R addresses begin C=AU;A=AU-PTT;P=EMAC, is served by another routing domain named EMAC. EMAC is represented as a Domain entity in the directory (see Figure 7–4).

```
CREATE MTS "/MTS=ACME" ORADDRESS "C=AU;A=AU-PTT;P=EMAC" -
    ROUTING INSTRUCTION [ACTION=TRANSFER TO DOMAIN, -
    SERVER DOMAIN="EMAC"]
```

8 These are entries for the O/R addresses:

C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=WELL
C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=AUCK

Routing information for these O/R addresses indicates a redirection O/R address for the Secretaries of the WELL and AUCK areas of the ACME Shoe Corporation. In this way, messages that are addressed to O/R addresses that include C=NZ;A=NZ-PTT;P=ACME;OU1=WELL or C=NZ;A=NZ-PTT;P=ACME;OU1=AUCK, and that have no more detailed O/R address attributes stored in the directory, can be delivered to the respective Secretary.

```
CREATE MTS "/MTS=ACME" ORADDRESS -
    "C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=WELL" -
    ROUTING INSTRUCTION [ACTION=MD REDIRECT, -
    REDIRECTION="C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=WELL;CN=WELL Secretary"]

CREATE MTS "/MTS=ACME" ORADDRESS -
    "C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=AUCK" -
    ROUTING INSTRUCTION [ACTION=MD REDIRECT, -
    REDIRECTION="C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=AUCK;CN=AUCK Secretary"]
```

Note that you also need to create an O/R address entry for the WELL and AUCK Secretaries as employees of the ACME Shoe Corporation.

9 This is the entry for the O/R address:
C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=WELL;CN=Clare Roberts

Routing information for this O/R address indicates that the O/R address is served by an MTA and User Agent. As the ACME routing domain is multihomed, and the User Agent is a registered Agent, a definitive O/R address is also required.

```
CREATE MTS "/MTS=ACME" ORADDRESS -
    "C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=WELL;CN=Clare Roberts" -
     ROUTING INSTRUCTION [ACTION=DELIVER, SERVER MTA="WELL.MTA-NODE6", -
     AGENT="UA1", DEFINITIVE ORADDRESS = -
     "C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=WELL;CN=Clare Roberts"]
```

In addition to the routing information held in the directory, you need an Agent entity for “UA1” that represents the routing information required by the MTA “WELL.MTA-NODE6”.

You can prevent Clare Roberts exchanging mail with individuals in routing domains in other X.400 management domains, as follows:

```
SET MTS "/MTS=ACME" ORADDRESS -
    "C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=WELL;CN=Clare Roberts" -
     MAY CROSS CCITT BOUNDARIES FALSE
```

Note that you also need to include the Different CCITT Domain attribute on domain entries in the directory; see Section 7.5.

10 This is the entry for the O/R address:

```
C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=AUCK;CN=William Laurence
```

Routing information for the entry indicates that the O/R address is served by an MTA and User Agent. As the ACME routing domain is multi-homed, and the User Agent is a registered Agent, a definitive O/R address is also required.

```
CREATE MTS "/MTS=ACME" ORADDRESS -
    "C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=AUCK;CN=William Laurence" -
       ROUTING INSTRUCTION [ACTION=DELIVER, SERVER MTA="AUCK.MTA-NODE1", -
       AGENT="UA2", DEFINITIVE ORADDRESS = -
       "C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=AUCK;CN=William Laurence"]
```

William Laurence is communicating with individuals in routing domains based on 1984 MHS standards and requires an attribute for the O/R address entry representing a Personal Name.

```
SET MTS "/MTS=ACME" ORADDRESS -
    "C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=AUCK;CN=William Laurence" -
    PERSONAL NAME "S=Laurence;G=William"
```

In addition to the routing information held in the directory you need an Agent entity for "UA2" that represents the routing information required by the MTA "AUCK.MTA-NODE1".

7.5 Routing Domain Information

This section describes the domain entries that represent the other routing domains to which MTAs in your routing domain can connect, and associated routing information that you need to store in the directory for the other routing domains. These domain entries are represented by Domain entities.

Figure 7–4 Directory Entries for Routing Domains

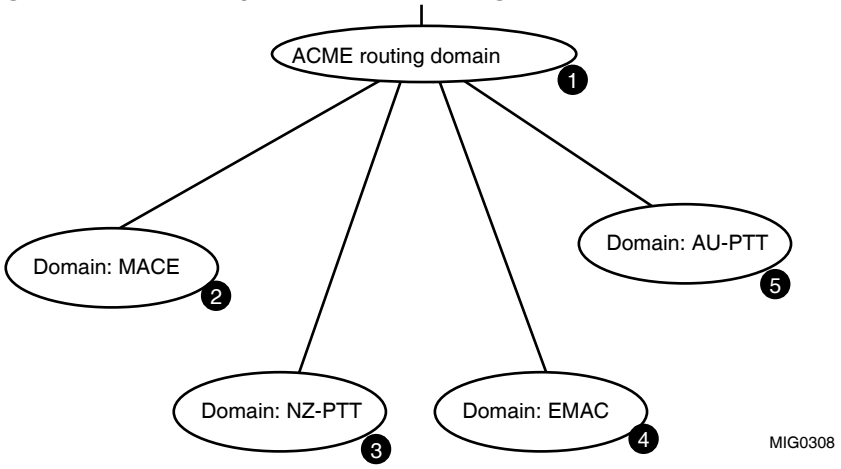


Figure 7–4 is an example of how some of the routing domains that the ACME Shoe Corporation connects to are represented in the directory. The numbers in the figure are:

1 ACME routing domain entry as specified in Section 7.4, step 1.

2 Entry for the routing domain connection to MACE, which is represented by the Domain entity with MACE as its identifier.

Routing information for this entry indicates that messages are routed to the MACE routing domain through the the boundary MTA "WELL.MTA-NODE6" (see Figure 4–5). The MACE routing domain is also part of another X.400 management domain.

```
CREATE MTS "/MTS=ACME" DOMAIN "MACE" -
    DIFFERENT CCITT DOMAIN TRUE, -
    ROUTING INSTRUCTION -
    [ACTION=TRANSFER TO DOMAIN, BOUNDARY MTA ="WELL.MTA-NODE6"]
```

3 Entry for the routing domain connection to NZ-PTT, which is represented by a Domain entity with NZ-PTT as its identifier.

Routing information for this entry indicates messages are routed to the NZ-PTT routing domain through the boundary MTA "AUCK.MTA-NODE5".

```
CREATE MTS "/MTS=ACME" DOMAIN "NZ-PTT" -
    ROUTING INSTRUCTION -
    [ACTION=TRANSFER TO DOMAIN, BOUNDARY MTA ="AUCK.MTA-NODE5"]
```

Note that you might need routing information for all the individuals within the NZ-PTT routing domain. This is not shown as an example in this guide.

4 Entry for the routing domain connection to EMAC, which is represented by a Domain entity with EMAC as its identifier. This routing domain is part of another X.400 management domain (see Figure 4–6).

Routing information for this entry indicates that messages are routed to the EMAC routing domain through another routing domain, MACE.

```
CREATE MTS "/MTS=ACME" DOMAIN "EMAC" -
    DIFFERENT CCITT DOMAIN TRUE, -
    ROUTING INSTRUCTION -
    [ACTION=TRANSFER THROUGH DOMAIN, SERVER DOMAIN="MACE"]
```

5 Entry for the routing domain connection to AU-PTT, which is represented by the Domain entity with AU-PTT as an identifier.

Routing information for this entry indicates that messages are routed to the AU-PTT routing domain through the boundary MTA “AUCK.MTA-NODE5”.

```
CREATE MTS "/MTS=ACME" DOMAIN "AU-PTT" -
    DIFFERENT CCITT DOMAIN TRUE, -
    ROUTING INSTRUCTION -
    [ACTION=TRANSFER TO DOMAIN, BOUNDARY MTA ="AUCK.MTA-NODE5"]
```

Note that you need routing information for all the individuals within the AU-PTT routing domain. This is not shown as an example in this guide.

In addition to the routing information held in the directory, you require routing information at any boundary MTAs that are specified in routing instructions for particular entries. This routing information specifies how these MTAs make connections to peer MTAs in the other routing domain and is held locally at the boundary MTA in a Peer MTA entity, an entity of the MTA module.

The commands required to create a Peer MTA entity to complete the routing information for the MACE routing domain are:

1. `CREATE MTA PEER MTA [TYPE = MANUALLY CONFIGURED, -`
   `NAME = "MACE-MTA1"]`

   Use this command to create a Peer MTA entity of type manually configured with the name MACE-MTA1. The entity holds information about the MTA in the MACE routing domain to which the boundary MTA WELL.MTA-NODE6 in the ACME routing domain connects.

2. `SET MTA PEER MTA [TYPE = MANUALLY CONFIGURED, -`
   `NAME = "MACE-MTA1"] -`
   `APPLICATION CONTEXT MTS TRANSFER PROTOCOL 1984`

   Use this command to assign an attribute to the Peer MTA entity, which indicates that the peer MTA wants to use the application context MTS TRANSFER PROTOCOL 1984.

3. `SET MTA PEER MTA [TYPE = MANUALLY CONFIGURED, -`
   `NAME = "MACE-MTA1"] SESSION ADDRESS -`
   `"//""MACE-MTA1""/NS+1234ab01234c0de01234,CLNS"`

   Use this command to assign an attribute to the Peer MTA entity, which specifies the Session address for the peer MTA. Refer to the appropriate DECnet-Plus documentation for details of the syntax of the Session address.

4. `SET MTA PEER MTA [TYPE = MANUALLY CONFIGURED, -`
   `NAME = "MACE-MTA1"] -`
   `DIALOGUE MODE MONOLOGUE`

Use this command to assign an attribute to the Peer MTA entity, which specifies the dialogue mode for the connection between the peer MTA and the boundary MTA.

5.
```
SET MTA PEER MTA [TYPE = MANUALLY CONFIGURED, -
   NAME = "MACE-MTA1"] -
   LOCAL NAME "AUCK.MTA-NODE6" -
   LOCAL PASSWORD [TYPE=IA5, IA5 STRING="CHEESE"]
```

   Use this command to assign attributes to the Peer MTA entity, which specify a name and password that the boundary MTA, WELL.MTA-NODE6, will quote when connecting to the peer MTA.

6.
```
SET MTA PEER MTA [TYPE = MANUALLY CONFIGURED, -
   NAME = "MACE-MTA1"] -
   PEER NAME "MACE-MTA1" -
   PEER PASSWORD [TYPE=OCTET, OCTET STRING='4241434F4E'H] -
   PEER DOMAIN "MACE"
```

   Use this command to assign attributes to the Peer MTA entity, which specify the name and password that the peer MTA, MACE-MTA1, will quote when making a connection to the boundary MTA, WELL.MTA-NODE6. The command also assigns the attribute that specifies the name of the routing domain where the peer MTA is located. Note that this routing domain name must match the name of the Domain entity where the boundary MTA is specified as part of the routing instruction. In this case the name of the routing domain is MACE.

7.
```
SET MTA PEER MTA [TYPE = MANUALLY CONFIGURED, -
   NAME = "MACE-MTA1"] -
   TEMPLATE NAME ("MTA_CLNS") -
   TRANSPORT SERVICE OPTIONS (OSI)
```

   Use this command to assign the name of the Transport template and the transport service that will be used when the boundary MTA makes connections to the peer MTA.

   Note

   The Transport Service Options attribute is only available on Tru64 UNIX.

   In this case, the MTA makes connections to the peer MTA using a Template entity describing transport characteristics for a Connectionless Network Service (CLNS), using OSI transport.

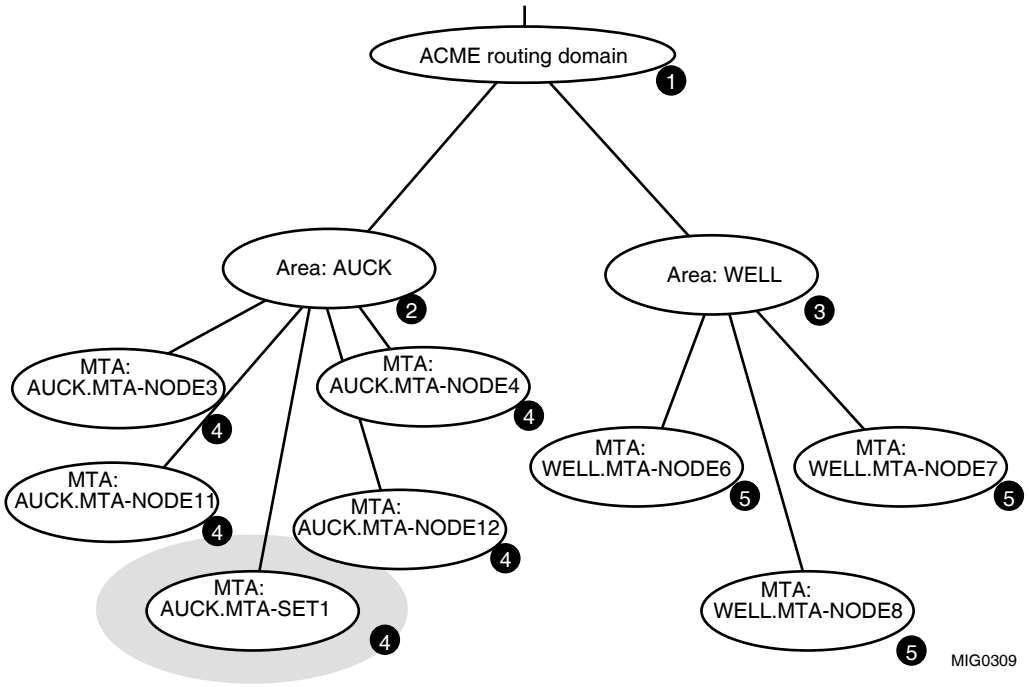
Add the commands to create a Peer MTA entity to the MTA's startup script. This ensures that the Peer MTA entity is created whenever the node where the MTA is running is rebooted.

7.6 Area and MTA Information

This section describes the entries for areas and MTAs in the directory. It also describes the associated routing information. Areas are represented by Area entities and MTAs are represented by MTA entities. Both are entities of the MTS module.

Figure 7–5 is an example of how the areas and MTAs within the ACME Shoe Corporation are represented in the directory.

Figure 7–5 Directory Entries for Areas and MTAs



The numbers in the figure are:

1 ACME routing domain entry, as described in Section 7.4, step 1.

2 Entry for the AUCK area of the ACME routing domain, represented by an Area entity with AUCK as an identifier.

Attributes of this entry specify the name of the MTA that serves the area.

If the SERVER MTA attribute is omitted, area routing is not active for the specified area.

```
CREATE MTS "/MTS=ACME" AREA "AUCK"
SET MTS "/MTS=ACME" AREA "AUCK" SERVER MTA "AUCK.MTA-SET1"
```

3 Entry for the WELL area of the ACME routing domain, represented by an Area entity with WELL as its identifier.

Attributes of this entry specify the name of the MTA that serves the WELL area.

If the SERVER MTA attribute is omitted, area routing is not implemented for the specified area.

```
CREATE MTS "/MTS=ACME" AREA "WELL"
SET MTS "/MTS=ACME" AREA "WELL" SERVER MTA "WELL.MTA-NODE7"
```

4 Entries for the MTAs within the AUCK area: AUCK.MTA-NODE3, AUCK.MTA-NODE4, AUCK.MTA-NODE11, and AUCK.MTA-NODE12. These entries are represented by MTA entities that have the MTA names as their identifiers (see Figure 4–3). These entries and their associated attributes, for example, address and password, are set up as part of an individual MTA's setup procedure after the MTA has been installed on a particular node.

The MTA in the shaded area represents an MTA set, AUCK.MTA-SET1.

```
CREATE MTS "/MTS=ACME" MTA "AUCK.MTA-SET1" TYPE MTA SET
SET MTS "/MTS=ACME" MTA "AUCK.MTA-SET1" -
   MEMBERS {"AUCK.MTA-NODE11","AUCK.MTA-NODE12"}
```

5 Entries for the MTAs within the WELL area: WELL.MTA-NODE6, WELL.MTA-NODE7, and WELL.MTA-NODE8. These entries are represented by MTA entities that have the MTA names as their identifiers.

These MTA entries and their associated attributes, for example, address and password, are set up as part of an individual MTA's setup procedure once the MTA has been installed.

7.7 Foreign Address Information

You do not need to represent foreign mail addresses unless your routing domain connects to another routing domain through a Gateway that supports using the directory to translate user addresses, for example, the HP MAILbus 400 SMTP Gateway. Contact the person responsible for managing the Gateway to find out whether the Gateway does support address translation. If it does, you need to find out the format of the foreign addresses that you need to store

in the directory, and whether the Gateway is set up to translate complete or partial mail addresses.

If the Gateway does not use address translation, you do not need to read this section.

You need to create Foreign Address entities for each complete or partial foreign mail address that you want to store in the directory. For each foreign mail address there is an equivalent complete or partial O/R address. Each Foreign Address entity is identified by a foreign mail address. Refer to the documentation supplied with the Gateway for details of the addressing formats used in each of the routing domains served by the Gateway.

The Translation attribute of the Foreign Address entity specifies the O/R address translation for the foreign mail address represented by the Foreign Address entity.

Figure 7–6 Directory Entries for Foreign Addresses

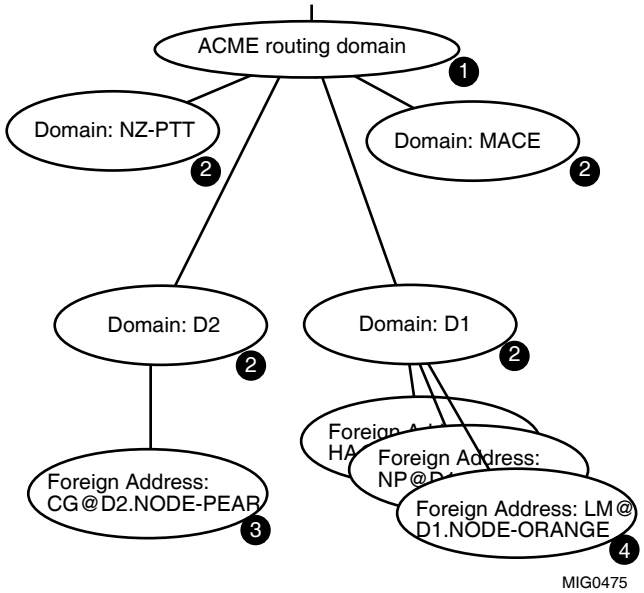


Figure 7–6 is an example of how an entry for a Foreign Address for the ACME Shoe Corporation is represented in the directory. The numbers in the figure are explained in the following list:

1 Entry for the ACME routing domain as specified in Section 7.4, step 1.

2 Entries for some of the routing domains that ACME connects to. You create entries for these routing domains and associated routing information as described in Section 7.5.

3 Entry for the foreign address, CG@D2.NODE-PEAR.

The following is an example of the entry for this foreign address with the equivalent O/R address translation:

```
CREATE MTS "/MTS=ACME" DOMAIN "D2" FOREIGN ADDRESS -
    "cg@d2.node-pear" TRANSLATION -
    "C=NZ; A=NZ-PTT; P=D2; O=D2; CN=Candide Gesser", -
    ROUTING INSTRUCTION -
    [ACTION=DELIVER, SERVER MTA="MTA-NODE13", AGENT="SMTP-GW"]
```

Candide Gesser also requires an O/R address entry and Routing Instruction.

```
CREATE MTS "/MTS=ACME" ORADDRESS "C=NZ;A=NZ-PTT;P=D2"
CREATE MTS "/MTS=ACME" ORADDRESS "C=NZ;A=NZ-PTT;P=D2;O=D2"
CREATE MTS "/MTS=ACME" ORADDRESS -
    "C=NZ;A=NZ-PTT;P=D2;O=D2;CN=Candide Gesser" -
    ROUTING INSTRUCTION [ACTION=TRANSFER TO DOMAIN, -
    SERVER DOMAIN = "D2"]
```

Specify the appropriate address translation for Candide Gesser.

```
SET MTS "/MTS=ACME" ORADDRESS -
     "C=NZ;A=NZ-PTT;P=D2;O=D2;CN=Candide Gesser" -
     TRANSLATIONS {[DOMAIN="D2", FOREIGN ADDRESS="cg@d2.node-pear"]}
```

Note that the routing information can also be provided at the partial O/R address “C=NZ;A=NZ-PTT;P=D2;O=D2”, and apply to the entire user community in routing domain “D2”.

4 These are the entries for the following foreign addresses:

HA@D1.NODE-PLUM
NP@D1.NODE-BANANA
LM@D1.NODE-ORANGE

The following are examples of the entries for these foreign addresses with their equivalent O/R addresses:

```
CREATE MTS "/MTS=ACME" DOMAIN "D1" FOREIGN ADDRESS -
    "ha@d1.node-plum" TRANSLATION -
    "C=NZ;A=NZ-PTT;P=D1;O=D1;CN=Harriet Anderson"

CREATE MTS "/MTS=ACME" DOMAIN "D1" FOREIGN ADDRESS -
    "np@d1.node-banana" TRANSLATION -
    "C=NZ;A=NZ-PTT;P=D1;O=D1;CN=Nigel Potts"

CREATE MTS "/MTS=ACME" DOMAIN "D1" FOREIGN ADDRESS -
    "lm@d1.node-orange" TRANSLATION -
    "C=NZ;A=NZ-PTT;P=D1;O=D1;CN=Laura Mundy"
```

These individuals also need O/R address entries with associated routing information. The routing information can be placed at the partial O/R address entry representing the community of users in the routing domain D1.

```
CREATE MTS "/MTS=ACME" ORADDRESS "C=NZ;A=NZ-PTT;P=D1"
CREATE MTS "/MTS=ACME" ORADDRESS "C=NZ;A=NZ-PTT;P=D1;O=D1" -
    ROUTING INSTRUCTION [ACTION=TRANSFER TO DOMAIN, -
    SERVER DOMAIN = "D1"]
```

Note that you can also include the routing information at the complete O/R addresses.

In addition, if these individuals want to receive replies to their messages sent to the other routing domain, they will need translation information on their O/R address entries:

```
CREATE MTS "/MTS=ACME" ORADDRESS -
     "C=NZ;A=NZ-PTT;P=D1;O=D1;CN=Harriet Anderson" -
     TRANSLATIONS {[DOMAIN="D1", FOREIGN ADDRESS="ha@d1.node-plum"]}

CREATE MTS "/MTS=ACME" ORADDRESS -
    "C=NZ;A=NZ-PTT;P=D1;O=D1;CN=Nigel Potts" -
     TRANSLATIONS {[DOMAIN="D1", FOREIGN ADDRESS="np@d1.node-banana"]}

CREATE MTS "/MTS=ACME" ORADDRESS -
    "C=NZ;A=NZ-PTT;P=D1;O=D1;CN=Laura Mundy" -
     TRANSLATIONS {[DOMAIN="D1", FOREIGN ADDRESS="lm@d1.node-orange"]}
```

7.8 What to Do Next

You now need to set up your routing domain. This includes installing individual MTAs. Part III describes setting up your routing domain.

Part III

Setup

This part describes how to set up your routing domain based on the information in *Part II, Planning*.

The part comprises the following:

- Chapter 8, which describes setting up your MTAs on a Tru64 UNIX system.
- Chapter 9, which describes setting up your MTAs on OpenVMS VAX and Alpha systems.
- Chapter 10, which describes the routing information that you need to add when you have set up an MTA.
- Chapter 11, which describes how to make sure Agents using the XAPI interface and Shared File interface can communicate with the MTA.
- Chapter 12, which describes how you can use the mtamail User Agent supplied with the MTA.

One aspect of setting up your routing domain is installing the components of the MAILbus 400 MTA. Make sure that you have a copy of the installation documentation available when setting up your routing domain.

Note that you can choose to install all the components of the MAILbus 400 MTA throughout your routing domain before setting any of them up.

Have available the documentation for any Agents that you intend to use with the MAILbus 400 MTA.

8

Setting Up Your Routing Domain on a Tru64 UNIX System

This chapter describes how to use the information planned in the previous chapters to set up your MTAs on Tru64 UNIX systems.

The chapter is divided into the following areas:

- Upgrading from a previous version of the MTA.

  This assumes that an earlier version of the MTA was operating on the system and that the current directory configuration is still applicable.

- Setting up MTAs for the first time.

  Setting up an MTA for the first time is divided into two sections. The first describes setting up the first MTA within the routing domain, and the other section describes setting up subsequent MTAs in your routing domain.

  There are two sections because there are some setup tasks that only need to be done once within the routing domain. Other tasks need to be done on every system where you set up an MTA.

  - Setting up the first MTA in your routing domain.

    If you have read *MAILbus 400 Getting Started* you will already have set up your first MTA. However, you might want to read this section if you have used *MAILbus 400 Getting Started* to set up a pilot system, or if you want more explanation of the tasks that you have completed in *MAILbus 400 Getting Started*.

    Setting up your first MTA describes how to set up a single MTA communicating with a single DSA. For the first MTA that you set up you need to create the initial routing domain entries in the directory. These entries are then accessible to all the MTAs that you set up within your routing domain.

After you have set up the first MTA you can either set up the remaining routing information for your routing domain, for example, O/R addresses, or set up more MTAs in your routing domain.

– Setting up MTAs on other systems in your routing domain.

  This describes how to set up MTAs other than the first MTA. Some of the routing information for these MTAs will already exist, the rest will need to be set up.

  You might decide that you want a particular MTA to contact a different DSA for routing information. In this chapter you will set up all your MTAs to contact the same DSA. When the MTAs have been set up successfully, you can then choose to modify the MTA setup such that an MTA can contact another DSA of your choice. In this way, you can confirm that the MTA setup is correct, before using alternative DSAs.

  If you decide to replicate the MTA routing information to multiple DSAs, that is, to shadow DSAs, you must make sure that each DSA holds an up-to-date copy of the MAILbus 400 MTA's routing information. The MTA's directory requirements are described in Chapter 6.

Note

Register a valid MAILbus 400 MTA license for each MTA that you set up.

Some of the commands used to set up the MTA are NCL commands and some are Tru64 UNIX system commands. In all cases, enter the commands from the superuser account.

8.1 Tasks to Complete Before Setting Up any MTAs

Make sure that you are working from a privileged account on the system where you intend to set up the MTA.

8.1.1 Ensure the Directory Server Subset is Installed

The DSA must already be installed within your routing domain. This DSA might be on the same system where you intend to set up the MTA, or it might be on a different system. If the DSA is on a different system, you need to know the name of the system where it is running. The HP Enterprise Directory Service installation documentation describes how to install the HP Enterprise Directory Service Server subset.

8.1.2 Ensure the Directory Base Subset is Installed

Make sure that the HP Enterprise Directory Service Base subset is installed on the system where you intend to set up the MAILbus 400 MTA. The HP Enterprise Directory Service installation documentation describes how to install the HP Enterprise Directory Service Base subset.

Later sections in this chapter explain how to configure the HP Enterprise Directory Service Base subset for the MTA.

8.1.3 Ensure the Required MTA Subsets are Installed

Follow the instructions in *HP MAILbus 400 MTA Installing on a Tru64 UNIX System* and install the following subsets on each system in your routing domain where you intend to operate an MTA:

MAILbus 400 MTA Base
MAILbus 400 MTA Mgt
MAILbus 400 MTA Server

When you install the MAILbus 400 MTA Server subset, the HP DECnet-Plus startup file /sbin/osi_applstartup is updated to include the appropriate MTA startup commands. The HP DECnet-Plus installation makes sure that /sbin/osi_applstartup is executed at system startup.

8.1.4 What to Do Next

If you are upgrading from a previous version of the MTA, continue at Section 8.2.

If you are setting up MTAs in the routing domain for the first time, continue at Section 8.3. Note that you are advised to follow the instructions in *MAILbus 400 Getting Started* to set up your first MTA.

If you have already set up the first MTA and are now setting up MTAs on other systems in the routing domain, continue at Section 8.4.

8.2 Upgrading from a Previous Version of the Product

This section describes the tasks that you need to complete when upgrading from a previous version of the MTA. If you are not upgrading, but setting up MTAs for the first time, continue at Section 8.3.

If you want to manage your routing information from another system, but not operate an MTA, you can set up the management subset of the MTA without the Server (see Section 8.9).

8.2.1 Check That There is a DUA Defaults File on the System

As you are upgrading from a previous version of the MAILbus 400 MTA, connections to the HP Enterprise Directory Service from this system will already be set up. To confirm this, check that there is a DUA defaults file located at /etc/dua.defaults.

If there is a dua.defaults file, you can continue at Section 8.2.2. If there is not, you need to configure the HP Enterprise Directory Service Base subset before continuing at Section 8.2.2. Section 8.3.1 describes how to configure the HP Enterprise Directory Service Base subset.

8.2.2 Add Your Customizations to the New Startup File

Update the newly installed MTA startup script and other template scripts, where appropriate, from the customized versions you were using before you installed the new version of the product. Add your customizations to the new scripts before setting up or starting the MTA. The new start_mta.ncl script contains new commands so it is important that you add your customizations to the new scripts and that you do not delete the new script. In particular, you must add the Create MTA entry from your previous script.

When you edit start_mta.ncl to create the Peer MTA and Agent entities, you are advised to create individual NCL scripts and add the "do" commands to execute the scripts in start_mta.ncl. Chapter 10 provides examples of how you do this.

Continue at Section 8.2.3, which describes how to run the MTA Setup procedure.

8.2.3 Run the MTA Setup Procedure

This section describes how to use the MTA Setup procedure to set up an MTA.

Complete the following steps to set up an MTA that you have installed:

1. From an account with superuser privileges, execute the following script:

   ```
   # /var/mta/scripts/mta_setup
   ```

2. The procedure asks a number of questions. If you have executed the setup procedure before, your previous settings are offered as defaults. You can request Help for any question by responding to the question with a question mark (?).

   a. `Do you want to continue with this procedure? [yes]`

      If, after reading the overview of what the MTA Setup procedure will do, you decide not to continue, enter **No** and exit from the MTA Setup procedure; otherwise, enter **Yes**.

b. `What is the distinguished name of your routing domain?`

   Supply the distinguished name of the routing domain. This is the name that you used to create the MTS entity.

c. `What is the name of the MTA?`

   Supply the name of the MTA, including any area identifiers.

   The MTA name must be unique within the routing domain and be in the form: area.mta_name.

d. `What is the MTA password?`

   Supply a password for the MTA. The password can be from 1 to 62 characters from the printable string character set. The password is not displayed as you type it. Note that you must not enter a null password.

The MTA Setup procedure writes the command to create the MTA entity into the file /var/mta/scripts/start_mta.ncl. The MTA Setup procedure also creates the file /var/mts/scripts/mts_create_mta_entry.ncl, which contains the NCL commands to create the entry for the MTA in the directory.

The MTA Setup procedure takes all the NSAPs that it can detect on the system where the setup procedure is executed and adds them to the MTA's Presentation and Session addresses. However, in the case of RFC 1006, if

the MTA cannot determine the TCP/IP address from either /etc/hosts, or from the BIND server, you will need to add it manually to the MTA Presentation and Session addresses. If the MTA cannot find a TCP/IP address, you will see the following informational message:

```
No TCP/IP address for this system could be determined from /etc/hosts or
the BIND server.  Therefore this procedure cannot add an RFC1006 NSAP to
the mts_create_mta_entry.ncl script.
```

If you see this message, ask the person responsible for managing the system to find out the TCP/IP address for the system, and edit /var/mts/scripts/mts_create_mta_entry.ncl to include the RFC 1006 NSAP, before continuing at Section 8.2.4.

As an example, if you have the TCP/IP address 1.2.3.4, the NSAP is:

```
RFC1006+1.2.3.4+102,RFC1006
```

The following is an example of a CLNS and RFC 1006 NSAP within an MTA's Presentation address:

```
""MTA""/""MTA""/""MTA88-NODE1""/NS+4900AB01002CD03E45521,CLNS|
  RFC1006+1.2.3.4+102,RFC1006"
```

In this example, "102" is the TCP port used by the MAILbus 400 MTA. *HP MAILbus 400 MTA Tuning and Problem Solving* provides more information about TCP/IP addresses when used by the MTA.

8.2.4 Create the MTA Entry in the Directory

When you are satisfied that all the required NSAPs are included in the MTA's Presentation and Session addresses, create the MTA entry in the directory. Execute mts_create_mta_entry.ncl, which creates an MTA entity as a subentity of the MTS entity, as follows:

```
ncl> do /var/mts/scripts/mts_create_mta_entry.ncl
```

If this command fails, check:

- That the routing information is accessible to the master DSA as follows:

  ```
  NCL> show node "dsa_node" -
  _NCL> dsa naming context "/mts=routing-domain-name"
  ```

  If the DSA indicates that no object exists, the DSA does not have knowledge of the naming context for the routing domain. Contact the person responsible for managing the directory to find out what the problem is.

- That the system is authorized to manage the routing information.

  It is possible that the password attribute for the MTS entity has been modified since you last entered information in the directory.

8.2.5 Start the MTA

To start the MTA, execute start_mta.ncl using NCL as follows:

```
ncl> do /var/mta/scripts/start_mta.ncl
```

The MTA is now operational.

If the MTA fails to start, check that the create MTA command has been added to the MTA Startup script correctly (see Section 8.2.2).

8.2.6 Checking create_mta_cons_templates.ncl

If this MTA can connect to peer MTAs either within its routing domain, or in other routing domains using CONS, you need to check that the network described by the X.25 Template is accessible to this MTA. This is the network referenced by the OSI Transport template used by the MTA. If the network is not accessible to the MTA, update the MTA's create_mta_cons_templates.ncl script to provide an alternative X.25 Template.

To change the X.25 Template referenced by the MTA, you must edit create_mta_cons_templates.ncl in /var/mta/scripts for this MTA and identify an alternative X.25 Template. This X.25 Template must describe an X.25 network with which this MTA can directly communicate. If you do not have a suitable X.25 Template defined on your system, the commands to create one are included in the create_mta_cons_templates.ncl script. Follow the instructions specified in the file to create a suitable template.

8.2.7 What to Do Next

You have now completed the upgrade tasks for the MTA. You can continue at Section 8.5, which describes how to run the MTA verification procedure.

8.3 Setting Up an MTA in the Routing Domain for the First Time

This section describes the tasks that you need to complete when setting up the MTA for the first time within this routing domain. If you are upgrading from a previous release of the MTA, you do not need to read this section and should refer to Section 8.2.

If you have already set up one MTA and want to continue and set up more MTAs within the routing domain, refer to Section 8.4.

8.3.1 Set Up Access to a DSA

This section describes what you need to do to make sure that you are able to create the routing domain entry for your routing domain in the HP Enterprise Directory Service. This includes:

- Configuring the DSA.
- Creating a naming context, and possibly a Subordinate Reference entity, for the MTA's routing information at the DSA.
- Creating a DUA defaults file.

8.3.1.1 Configure the DSA

On the system where you have installed the DSA, complete the following:

- Execute the following script:

  ```
  # /var/dxd/scripts/dsa_configure
  ```

  If a DSA is already created and running on this node, an informational message is displayed stating that a DSA already exists.

- Start the DSA as follows:

```
NCL> create dsa
NCL> enable dsa
```

If NCL returns errors for any of these commands, refer to the HP Enterprise Directory Service documentation for information on how to solve the problem.

8.3.1.2 Create a Naming Context for the MTA

Chapter 6 describes the MTA's directory requirements, in particular, whether a Subordinate Reference entity is required. If, after reading Chapter 6, you decide that you need a Subordinate Reference entity, refer to the HP Enterprise Directory Service documentation, which describes Subordinate Reference entities and how to create them in detail.

Note that you require write-access to the directory in order to create any entries in the directory.

The Subordinate Reference entity must exist at the system that holds the master copy of the Naming Context superior to the Naming Context that you want to create for your routing information.

Create a Naming Context entity for your routing information on the DSA that will hold the master copy of the MTA's routing information as follows:

```
ncl> create dsa naming context "/mts=routing-domain-name"
```

The DSA where you create the Naming Context entity is referred to as the **master DSA** for your routing domain.

If you receive an error indicating there is no parent entry for the Naming Context entity you are trying to create, there is a superior Naming Context defined that you are unaware of. Contact the person responsible for managing the directory, or refer to *HP Enterprise Directory - Problem Solving* for information about solving problems with Naming Contexts.

8.3.1.3 Create the DUA Defaults File

Complete the following to create a DUA defaults file on the node where you intend to set up this MTA:

1. Check that the DSA is available as follows:

   ```
   ncl> show node "dsa_node" dsa state
   ```

   where ***dsa_node*** is the name of the system where the master DSA for the routing information is installed.

   The DSA state must be ON. If the DSA is not ON, contact the person responsible for managing the DSA to find out whether there is a problem. You will not be able to continue until the DSA is ON.

2. Create a DUA defaults file.

   The MTA uses the information in the HP Enterprise Directory Service DUA defaults file to access the DSA.

   Execute dua_configure as follows and, when prompted, provide the name of the system where the master DSA is located:

   ```
   # /var/dxd/scripts/dua_configure
   ```

   dua_configure creates the file /etc/dua.defaults, which contains the Presentation address of the DSA. This address is then used by all the directory applications on the system, including the MAILbus 400 MTA, to access the DSA.

If dua_configure fails to contact the master DSA, or fails to obtain information from the DSA, it displays an error message, and asks whether you want to try a different DSA. Answer **No**, because at this point you must use the master DSA where the Naming Context entity for the routing domain is created.

If the utility continues to fail, refer to *HP Enterprise Directory - Problem Solving* for information about problems using the DUA configuration utility.

8.3.2 Create the Routing Domain and Area Entries for the MTA

This section describes how to create the MTA's routing domain entry, and, where applicable, the Area entries in the directory.

Creating the routing domain entry comprises:

- Creating the MTS entity and setting the MTS entity password
- Adding global domain identifiers

You can then add Area entities where required.

You cannot create these entries unless a Naming Context entity already exists (see Section 8.3.1.2.

8.3.2.1 Create the MTS Entity for Your Routing Domain

This section describes how to create the MTS entity that represents your routing domain and set the MTS entity password.

If you do not have write-access to the directory, you will also need to quote values for the Authorizing User and Authorizing Password arguments when you create the MTS entity. If you have installed the directory solely for the MTA, you will not need authority as the directory is installed with write-access for all users. In this case, you can omit the Authorizing User and Authorizing Password arguments.

Create the MTS entity as follows:

```
ncl> create mts "/mts=routing-domain-name" -
_ncl> password "mts-password", -
_ncl> authorizing user "directory-user", -
_ncl> authorizing password "directory-pass"
```

where:

- ***mts-password*** is the password that you want to assign to your routing domain.

  Chapter 7 describes why a password is required.

- ***directory-user*** is the distinguished name of a user who has write-access to the directory.

- ***directory-pass*** is the Password attribute of the user with distinguished name ***directory-user***.

If NCL returns an error when you enter this command, check that you have used the correct syntax for the command. The *MTS Module Online Help* describes the syntax for the Create MTS command in detail.

Do not forget the password that you enter when you create the MTS entity. Later sections in this chapter describe how you use this password to ensure that you can manage the MTA routing information from other systems in the routing domain.

8.3.2.2 Add Global Domain Identifiers

Add the list of GDIs for your routing domain to the routing domain entry:

```
ncl> set mts "/mts=routing-domain-name" -
_ncl> global domain identifiers -
_ncl> {"c=c-name;a=admd-name;p=prmd-name",...}
```

where:

- ***c-name*** is the value of the Country Name attribute for the GDI.

- ***admd-name*** is the value of the Administration Domain Name attribute for the GDI.

- ***prmd-name*** is the value of the Private Domain Name attribute for the GDI.

8.3.2.3 Create Area Entries

If you have not planned any areas for your routing domain, continue at Section 8.3.3.

If you have planned to divide your routing domain into areas, create Area entities for each area in your routing domain:

```
ncl> create mts "/mts=routing-domain-name" area "area-name"
```

where ***area-name*** is an area name.

If you have planned a hierarchy of areas, for example, area1.area2, create the most significant area in the hierarchy first. This is so that the area hierarchy is stored correctly in the directory. Create the Area entities for area1 and area2, where area2 is subordinate to area1, as follows:

```
ncl> create mts "/mts=routing-domain-name" area "area1"
ncl> create mts "/mts=routing-domain-name" area "area1.area2"
```

8.3.3 What to Do Next

Continue and run the MTA Setup procedure as described in Section 8.4.3.

8.4 Adding More MTAs to Your Routing Domain

This section assumes that one MTA has been set up and is operational. Complete this section for each additional MTA that you want to operate within your routing domain on another system.

If you are upgrading an MTA, refer to Section 8.2.

If you want to manage your routing information from a system where you are not operating an MTA Server, you can set up the management subset of the MTA only (see Section 8.9).

The DSA that an MTA will connect to must already be installed and set up. This DSA might be on the same system where you intend to set up the MTA, or it might be on a different system. If the DSA is on a different system, you need to know the name of the system where it is running.

In this section you will:

- Set up the MTA's access to the master DSA.

  When you have completed setting up the MTA and confirmed that the set up is successful, you can then choose to modify the MTA setup such that the MTA contacts a shadow DSA. This is described in Section 8.6.

- Run the MTA Setup procedure.

- Create MTA entries in the directory.

8.4.1 Set Up Access to a DSA

On the system where you have installed the MTA subsets, complete the following:

1. Check that the DSA is available as follows:

   ```
   ncl> show node "dsa_node" dsa state
   ```

   where ***dsa_node*** is the name of the system where the master DSA for the routing domain is installed.

   The DSA state must be ON. If the DSA is not ON, contact the person responsible for managing the DSA to find out whether there is a problem. You will not be able to continue until the DSA is ON.

2. Create a DUA defaults file.

   Execute dua_configure as follows and, when prompted, provide the name of the system where the master DSA is located:

   ```
   # /var/dxd/scripts/dua_configure
   ```

   dua_configure creates the file /etc/dua.defaults, which contains the Presentation address of the DSA. This address is then used by all the directory applications on the system, including the MAILbus 400 MTA, to access the DSA.

   If dua_configure fails to contact the master DSA, or fails to obtain information from the DSA, it displays an error message, and asks whether you want to try a different DSA. Answer **No** because at this point you must use the master DSA where the Naming Context entity for the routing domain is created.

   If the utility continues to fail, refer to *HP Enterprise Directory - Problem Solving* for information about problems using the DUA configuration utility.

3. Check that the DSA has knowledge of the naming context for the routing domain as follows:

   ```
   NCL> show node "dsa_node"-
   _NCL> dsa naming context "/mts=routing-domain-name"
   ```

   If you have set up your first MTA using the instructions in *MAILbus 400 Getting Started*, *routing-domain-name* is the same as *email-system-name* used in *MAILbus 400 Getting Started*.

If the DSA indicates that no such object exists, the DSA does not have knowledge of the naming context for the routing domain. In this case, contact the person responsible for managing the directory to find out what the problem is.

4. If the DSA does have knowledge of the naming context, check that the DSA also has knowledge of the routing information for your routing domain as follows:

   ```
   NCL> show mts "/mts=routing-domain-name"
   ```

   If the DSA does not have knowledge of the MTA's routing information, check that the DXD Transport Templates have been set up correctly. The HP Enterprise Directory Service documentation explains how to set up the DXD Transport Templates.

   When you have confirmed that the DSA has knowledge of your routing information, you can continue at Section 8.4.2.

8.4.2 Provide Authorization to Manage the MTA's Routing Information

Provide the necessary authorization so you can manage the MTA's routing information from this system. Use the following command to authorize the MTS entity:

```
ncl> authorize mts "/mts=routing-domain-name" password "mts-password"
```

where ***mts-password*** is the password that you provided when you created the routing domain entry as described in Section 8.3.2.1.

Continue at Section 8.4.3 and run the MTA Setup procedure.

8.4.3 Run the MTA Setup Procedure

This section describes how to use the MTA Setup procedure to set up an MTA.

Complete the following steps to set up an MTA that you have installed:

1. From an account with superuser privileges, execute the following script:

   ```
   # /var/mta/scripts/mta_setup
   ```

2. The procedure asks a number of questions. If you have executed the setup procedure before, your previous settings are offered as defaults. You can request Help for any question by responding to the question with a question mark (?):

   a. `Do you want to continue with this procedure? [yes]`

      If, after reading the overview of what the MTA Setup procedure will do, you decide not to continue, enter **No** and exit from the MTA Setup procedure; otherwise, enter **Yes**.

b. `What is the distinguished name of your routing domain?`

   Supply the distinguished name of the routing domain. This is the name that you used to create the MTS entity (the routing domain entry) in the directory in Section 8.3.2, for example, /mts=acme.

c. `What is the name of the MTA?`

   Supply the name of the MTA, including any area identifiers. The area identifiers are those specified in Section 8.3.2.3.

   The MTA name must be unique within the routing domain and be in the form: area.mta_name.

d. `What is the MTA password?`

   Supply a password for the MTA. The password can be from 1 to 62 characters from the printable string character set. The password is not displayed as you type it. Note that you must not enter a null password.

The MTA Setup procedure writes the command to create the MTA entity into the file /var/mta/scripts/start_mta.ncl. The MTA Setup procedure also creates the file /var/mts/scripts/mts_create_mta_entry.ncl, which contains the NCL commands to create the entry for the MTA in the directory.

The MTA Setup procedure takes all the NSAPs that it can detect on the system where the setup procedure is executed and adds them to the MTA's Presentation and Session addresses. However, in the case of RFC 1006, if the MTA cannot determine the TCP/IP address from either /etc/hosts, or from the BIND server, you will need to add it manually. If the MTA cannot find a TCP/IP address, you will see the following informational message:

```
No TCP/IP address for this system could be determined from /etc/hosts or
the BIND server.  Therefore this procedure cannot add an RFC1006 NSAP to
the mts_create_mta_entry.ncl script.
```

If you see this message, ask the person responsible for managing the system to find out the TCP/IP address for the system, and edit /var/mts/scripts/mts_create_mta_entry.ncl to include the RFC 1006 NSAP, before continuing at Section 8.4.3.1.

As an example, if you have the TCP/IP address 1.2.3.4, the NSAP is:

```
RFC1006+1.2.3.4+102,RFC1006
```

The following is an example of a CLNS and RFC 1006 NSAP within an MTA's Presentation address:

```
"""MTA""/""MTA""/""MTA88-NODE1""/NS+4900AB01002CD03E45521,CLNS|
  RFC1006+1.2.3.4+102,RFC1006"
```

In this example, “102” is the TCP port used by the MAILbus 400 MTA. *HP MAILbus 400 MTA Tuning and Problem Solving* provides more information about TCP/IP addresses when used by the MTA.

8.4.3.1 Create the MTA Entry in the Directory

When you are satisfied that all the MTA NSAPs are included in the MTA Presentation and Session addresses, create the MTA entry in the directory. Execute mts_create_mta_entry.ncl, which creates an MTA entity as a subentity of the MTS entity.

Execute mts_create_mta_entry.ncl using NCL as follows:

```
ncl> do /var/mts/scripts/mts_create_mta_entry.ncl
```

If this command fails, check:

- That you have created the appropriate MTS and Area entities in the directory (see Section 8.3.2).
- That the routing information is accessible to the master DSA as follows:

  ```
  NCL> show node "dsa_node" -
  _NCL> dsa naming context "/mts=routing-domain-name"
  ```

  If the master DSA indicates that no object exists, the DSA does not have knowledge of the naming context for the routing domain. Contact the person responsible for managing the the directory to find out what the problem is.
- That you have used the correct password to authorize this system.

  It is possible that the password has been modified since you last entered information using the MTS entity.

8.4.4 Start the MTA

To start the MTA, execute start_mta.ncl using NCL as follows:

```
ncl> do /var/mta/scripts/start_mta.ncl
```

The MTA is now operational.

8.4.5 Checking create_mta_cons_templates.ncl

If this MTA can connect to peer MTAs either within its routing domain, or in other routing domains using CONS, you need to check that the network described by the X.25 Template is accessible to this MTA. This is the network referenced by the OSI Transport template used by the MTA. If the network is not accessible to the MTA, update the MTA’s

create_mta_cons_templates.ncl script to provide an alternative X.25 Template. If you do not need to modify create_mta_cons_templates.ncl, continue at Section 8.5.

To change the X.25 Template referenced by the MTA, you must edit create_mta_cons_templates.ncl in /var/mta/scripts for this MTA and identify an alternative X.25 Template. This X.25 Template must describe an X.25 network with which this MTA can directly communicate. If you do not have a suitable X.25 Template defined on your system, the commands to create one are included in the create_mta_cons_templates.ncl script. Follow the instructions specified in the file to create a suitable template.

8.5 Using the MTA Verification Procedure

This section describes how to verify that the MTA is set up correctly.

Use the MTA verification procedure to check that your MTA installation and setup have been successful.

To run the MTA verification procedure, execute the following command on the system where the MTA is installed, from an account with superuser privileges:

```
# /usr/examples/mta/mta_vp.sh
```

If the MTA verification procedure fails for any reason, errors are written to /usr/examples/mta/mta_vp.log.

You can run the verification procedure at any time, provided the MTA is in the ON state, and the MTA is contacting the master DSA.

Note

If the MTA is contacting a shadow DSA, that is, a DSA that holds a copy of the MTA routing information, the procedure will fail.

The verification procedure confirms that a particular MTA is running correctly. The verification procedure confirms that:

- A message can be submitted from an Agent to the MTA.
- A message can be delivered back to the Agent from the MTA.
- The MTA can access the directory.

The following are examples of problems that the verification procedure can report:

- The MTA is not created and enabled.

- The node is not authorized to manage the routing information from the system.
- The DSA is not available.

You have now completed setting up this MTA.

8.6 Setting Up an MTA to Contact a Shadow DSA

This section describes how to set up your MTA to contact a shadow DSA. You must first make sure that you have set up the MTA as described in the earlier sections in this chapter.

Before you modify the MTA setup, make sure that you have replicated all the MTA's routing information to the shadow DSA (see the HP Enterprise Directory Service documentation).

On the system where the MTA that you want to modify is set up, complete the following:

1. Check that the DSA that you intend the MTA to contact is available as follows:

   ```
   ncl> show node "dsa_node" dsa state
   ```

   where ***dsa_node*** is the name of the system where the DSA is installed.

   The DSA state must be ON. If the DSA is not ON, contact the person responsible for managing the DSA to find out whether there is a problem. You will not be able to continue until the DSA is ON.

2. Shutdown the MTA as follows:

   ```
   # /var/mta/scripts/stop_mta.ncl
   ```

3. Create a new DUA defaults file.

   Execute dua_configure as follows and, when prompted, provide the name of the system where the DSA to which you intend the MTA to connect is located:

   ```
   # /var/dxd/scripts/dua_configure
   ```

   Executing this command will update the file /etc/dua.defaults, which contains the Presentation address of the DSA. This address is then used by all the directory applications on the system, including the MAILbus 400 MTA, to access the DSA.

If dua_configure fails to contact the DSA on the specified system, or fails to obtain information from the DSA, it displays an error message and asks whether you want to try a different DSA. Answer **No** and continue at step 6, and the MTA will continue to use the routing information at the master DSA.

4. Check that the DSA has knowledge of the naming context for the routing domain as follows:

```
NCL> show node "dsa_node" -
_NCL> dsa naming context "/mts=routing-domain-name"
```

   If the DSA indicates that no such object exists, the DSA does not have knowledge of the naming context for the routing domain. In this case, contact the person responsible for managing the directory to find out what the problem is.

5. If the DSA does have knowledge of the naming context, check that the DSA also has knowledge of the routing information for your routing domain as follows:

```
NCL> show mts "/mts=routing-domain-name"
```

   If the DSA does not have knowledge of the MTA's routing information, check that the MTA's routing information has been successfully replicated to the shadow DSA. In particular, check that the master and shadow DSAs "trust" each other. Refer to the HP Enterprise Directory Service documentation for details of replication and trust agreements.

   When you have confirmed that the DSA has knowledge of your routing information, you have completed the task of setting up your MTA to contact a shadow DSA. Make sure that you update the shadow DSA whenever you modify the MTA routing information in the directory.

6. Start the MTA as follows:

```
# /var/mta/scripts/start_mta.ncl
```

8.7 Keeping the MTA Startup Script Up-to-Date

You need to execute the MTA startup script every time you want to start the MTA. Changes made interactively to the MTA through NCL management are not automatically written to the MTA startup script file and will be lost each time the MTA is shut down. For this reason, it is important that you regularly update the MTA startup script and any other Peer MTA or Agent scripts to include any changes to the attributes of the MTA or its subentities that you make through NCL management.

8.8 What to Do Next

You have now completed setting up an MTA within your routing domain. You can now choose to:

- Upgrade more MTAs in your routing domain as described in Section 8.1 and Section 8.2.
- Set up more MTAs in your routing domain as described in Section 8.1 and Section 8.4.
- Set up the routing information for your routing domain:
  - Store the remainder of your routing information in the directory.

    This is information such as O/R addresses and information relating to other routing domains (see Section 10.1).
  - Create entities that represent peer MTAs locally at the MTA.

    These are Peer MTA entities and are described in Section 10.2.
  - Create entities that represent Agents locally at the MTA.

    These are Agent entities and are described in Section 10.3.

8.9 Setting Up Remote Access to the Routing Information in the Directory

This section describes how you set up a system to manage the MTA routing information in the directory, without setting up an MTA.

Before you set up remote access to your routing information on a system, make sure that you have installed the HP Enterprise Directory Service Base subset and the MAILbus 400 MTA Mgt subset.

Complete the following steps on the system where you have installed the required subsets:

1. Check that the DSA that you want to use is started. To do this, enter the following NCL command:

   ```
   ncl> show node "dsa_node" dsa state
   ```

   where ***dsa_node*** is the name of the system where the DSA to which you want to set remote access is installed.

   The DSA state must be ON. If the DSA is not ON, contact the person responsible for managing the DSA to find out whether there is a problem. You will not be able to continue until the DSA is ON.

2. Create a DUA defaults file.

   The MTA uses the directory DUA defaults file. If there is already a directory application installed on this system, there will be a DUA defaults file located at /etc/dua.defaults.

   If there is a DUA defaults file already set up on the system, continue at step 3.

   If no defaults file exists on the system, execute dua_configure and, when prompted, provide the name of the system where the DSA to which you want directory applications to connect is located:

   ```
   # /var/dxd/scripts/dua_configure
   ```

   dua_configure creates the file /etc/dua.defaults, which contains the Presentation address of the DSA. This address is then used by all the directory applications on the system.

   If dua_configure fails to contact the DSA on the specified system, or fails to obtain information from the DSA, it displays an error message, and asks whether you want to try a different DSA. You can name an alternative DSA that you know holds a copy of the routing information for your routing domain.

   If the utility continues to fail, refer to *HP Enterprise Directory - Problem Solving* for information about problems using the DUA configuration utility.

3. Provide the necessary authorization so you can manage the MTA's routing information from this system.

   Note that if this is an upgrade from a previous version of the MTA, the system will already be authorized. The exception to this is if the password has been modified during the upgrade.

   ```
   ncl> authorize mts "/mts=routing-domain-name" password "password"
   ```

   where ***password*** is the password that you provided when you created the routing domain entry.

9

Setting Up Your Routing Domain on an OpenVMS System

This chapter describes how to use the information planned in the previous chapters to set up your MTAs on an OpenVMS VAX or an OpenVMS Alpha system.

The chapter is divided into the following areas:

- Upgrading from a previous version of the MTA.

  This assumes that an earlier version of the MTA was operating on the system and that the current directory configuration is still applicable.

- Setting up MTAs for the first time.

  Setting up an MTA for the first time is divided into two sections. The first describes setting up the first MTA within the routing domain, and the other section describes setting up subsequent MTAs in your routing domain.

  There are two sections because there are some setup tasks that only need to be done once within the routing domain. Other tasks need to be done on every system where you set up an MTA.

  - Setting up the first MTA in your routing domain.

    If you have read *MAILbus 400 Getting Started* you will already have set up your first MTA. However, you might want to read this section if you have used *MAILbus 400 Getting Started* to set up a pilot system, or if you want more explanation of the tasks that you have completed in *MAILbus 400 Getting Started*.

    Setting up your first MTA describes how to set up a single MTA communicating with a single DSA. For the first MTA that you set up you need to create the initial routing domain entries in the directory. These entries are then accessible to all the MTAs that you set up within your routing domain.

After you have set up the first MTA you can either set up the remaining routing information for your routing domain, for example, O/R addresses, or set up more MTAs in your routing domain.

– Setting up MTAs on other systems in your routing domain.

  This describes how to set up MTAs other than the first MTA. Some of the routing information for these MTAs will already exist, the rest will need to be set up.

  You might decide that you want a particular MTA to contact a different DSA for routing information. In this chapter you will set up all your MTAs to contact the same DSA. When the MTAs have been set up successfully, you can then choose to modify the MTA setup such that an MTA can contact another DSA of your choice. In this way, you can confirm that the MTA setup is correct, before using alternative DSAs.

  If you decide to replicate the MTA routing information to multiple DSAs, that is, to shadow DSAs, you must make sure that each DSA holds an up-to-date copy of the MAILbus 400 MTA routing information. The MTA's directory requirements are described in Chapter 6.

Note

You should register a valid MAILbus 400 MTA license for each MTA that you set up in a cluster.

The chapter describes how you set up each component so that you have a fully operational routing domain. Some of the commands used to set up the MTA and directory are NCL commands and some are OpenVMS system commands. In all cases, you must enter the commands from a privileged account, for example, the SYSTEM account.

9.1 Setting Up the MTA in a Cluster

You can set up more than one MAILbus 400 MTA in a cluster. However, each MAILbus 400 MTA is independent, that is, the MAILbus 400 MTAs will not share workspaces and will operate as they would on individual nodes.

If you want to operate more than one MAILbus 400 MTA in a cluster, install the MAILbus 400 MTA once on each system disk in the cluster and then set up the MAILbus 400 MTA on each node in the cluster where you want it to run. Note however, that you can only set up the MTA on nodes that have access to the device where the [DXD$SERVER] directory for the HP Enterprise Directory Service is located. This is because directory applications, in this case, the MAILbus 400 MTA, obtain information about how to access the

HP Enterprise Directory Server from DXD$DUA_DEFAULTS.DAT, which is located in the [DXD$SERVER] directory.

9.2 Tasks to Complete Before Setting Up Any MTAs

Make sure that you are working from a privileged account on the system where you intend to set up the MTA.

9.2.1 Ensure the Directory Server Component is Installed

The DSA must already be installed. This DSA might be on the same system as the system where you intend to set up the MTA, or it might be on a different system. If the DSA is on a different system, you need to know the name of the system where it is running.

9.2.2 Ensure the Directory Base Component is Installed

Before starting to set up any MTAs, you need to make sure that the HP Enterprise Directory Service Base component is installed on the system where you intend to set up the MAILbus 400 MTA. The HP Enterprise Directory Service installation documentation describes how to install the HP Enterprise Directory Service Base component.

Later sections in this chapter explain how to set up the HP Enterprise Directory Service Base component for the MTA.

9.2.3 Ensure the Required MTA Components are Installed

Follow the instructions in the MTA installation documentation and install the following components on the system in your routing domain where you intend to operate the MTA:

MAILbus 400 MTA Base
MAILbus 400 MTA Mgt
MAILbus 400 MTA Server

9.2.4 What to Do Next

If you are upgrading from a previous version of the MTA, continue at Section 9.3.

If you are setting up MTAs in the routing domain for the first time, continue at Section 9.4. Note that you are advised to follow the instructions in *MAILbus 400 Getting Started* to set up your first MTA.

If you have already set up the first MTA and are now setting up MTAs on other systems in the routing domain, continue at Section 9.5.

9.3 Upgrading from a Previous Version of the Product

This section describes the tasks that you need to complete when upgrading from a previous version of the MTA. If you are not upgrading, but setting up MTAs for the first time (see Section 9.4).

If you want to manage your routing information from another system, but not operate an MTA, you can set up the management component of the MTA without the Server (see Section 9.10).

9.3.1 Check That There is a DUA Defaults File on the System

As you are upgrading from a previous version of the MAILbus 400 MTA, connections to the HP Enterprise Directory Service from this system will already be set up. To confirm this, check that there is DUA defaults file located at DXD$DIRECTORY:DXD$DUA_DEFAULTS.DAT.

If there is a DUA defaults file, you can continue at Section 9.3.2. If there is not, you need to configure the HP Enterprise Directory Service Base component before continuing at Section 9.3.2. Section 9.4.1 describes how to configure the HP Enterprise Directory Service Base component.

9.3.2 Initialize the MTS Process

You now need to initialize the MTS process required by the MAILbus 400 MTA:

```
$ @SYS$STARTUP:MTA$MTS_INIT
```

You only need to initialize the MTS process in this way the first time that you set up this MTA. At all other times, the MTS process is initialized within MTA$COMMON_STARTUP.COM.

Continue at Section 9.3.3, which describes how to run the MTA Setup procedure.

9.3.3 Run the MTA Setup Procedure

You must run the MTA Setup procedure to make sure that new copies of the MTA template files are copied to SYS$SPECIFIC from SYS$COMMON on this system.

This section describes how to use the MTA Setup procedure to set up an MTA.

Complete the following steps to set up an MTA on the node where you want the MTA to run:

1. Execute the following procedure:

   ```
   $ @SYS$STARTUP:MTA$SERVER_SETUP
   ```

2. The procedure asks a series of questions. If you have executed the setup procedure before, your previous settings are offered as defaults. You can request Help for any question by responding to the question with a question mark (?).

   a. `Do you want to continue with this procedure? [yes]`

      If, after reading the overview of what the MTA setup procedure will do, you decide not to continue, enter **No** and exit from the MTA setup procedure; otherwise, enter **Yes**.

   b. `What is the distinguished name of your routing domain?`

      Supply the distinguished name of the routing domain. This is the name that you used to create the MTS entity (the routing domain entry).

   c. `What is the name of the MTA?`

      Supply the name of the MTA, including any area identifiers.

      The MTA name must be unique within the routing domain and be in the form: area.mta_name.

   d. `What is the MTA password?`

      Supply a password for the MTA. The password can be from 1 to 62 characters from the printable string character set. Note that you must not enter a null password.

The MTA setup procedure enters the command to create the MTA entity into the script SYS$SPECIFIC:[SYS$STARTUP]MTA$START.NCL. The MTA setup procedure also creates the file MTA$MTS_CREATE_MTA_ENTRY.NCL in SYS$SPECIFIC:[SYS$STARTUP]. This file contains the NCL commands to create the entry for the MTA in the directory.

The MTA Setup procedure includes all the NSAPs that it can detect on the system where the setup procedure is executed in the MTA's Presentation and Session addresses, which is located at SYS$SPECIFIC:[SYS$STARTUP]MTA$MTS_CREATE_MTA_ENTRY.NCL. However, in the case of RFC 1006, if the MTA cannot determine the TCP/IP address, you will need to add it manually to MTA$MTS_CREATE_MTA_ENTRY.NCL in SYS$SPECIFIC:[SYS$STARTUP].

The MTA will only be able to determine the TCP/IP address automatically where the DEC TCP/IP Service for OpenVMS is installed. If the MTA cannot find a TCP/IP address, you will see the following informational message:

```
Cannot determine the RFC1006 NSAP of this system

If you require the MTA to operate over RFC 1006 then you must manually
edit the file SYS$SPECIFIC:[SYS$STARTUP]MTA$MTS_CREATE_MTA_ENTRY.NCL
and add the RFC 1006 NSAP to the MTA's Presentation and Session
addresses.

The syntax for an RFC1006 NSAP is:
 "RFC1006+<IP address or hostname>,RFC1006" for example
 "RFC1006+1.2.3.4,RFC1006"
```

The following is an example of a CLNS and RFC 1006 NSAP within an MTA's Presentation address:

```
"""MTA""/""MTA""/""MTA88-NODE1""/NS+4900AB01002CD03E45521,CLNS|
  RFC1006+1.2.3.4+102,RFC1006"
```

In this example, "102" is the TCP port used by the MAILbus 400 MTA. *HP MAILbus 400 MTA Tuning and Problem Solving* provides more information about TCP/IP addresses when used by the MTA.

9.3.4 Add Your Customizations to the New Startup File

Update the newly set up MTA startup script and other template scripts, where appropriate, from the customized versions you were using before you installed the new version of the product. Add your customizations to the new scripts before starting the MTA. The new MTA$START.NCL script contains new commands so it is important that you add modifications to the new scripts and that you do not delete the new script. In particular, you must add the Create MTA entry from your previous script.

When you edit start_mta.ncl to create the Peer MTA and Agent entities, you are advised to create individual NCL scripts and add the "do" commands to execute the scripts in MTA$START.NCL. Chapter 10 provides examples of how you do this.

9.3.5 Create the MTA Entry in the Directory

When you are satisfied that all the required NSAPs are included in the MTA's Presentation and Session addresses, create the MTA entry in the directory. Execute MTA$MTS_CREATE_MTA_ENTRY.NCL, which creates an MTA entity as a subentity of the MTS entity as follows:

```
NCL> DO SYS$STARTUP:MTA$MTS_CREATE_MTA_ENTRY.NCL
```

If this command fails, check:

- That the routing information is accessible to the master DSA as follows:

  ```
  NCL> SHOW NODE "dsa_node" -
  _NCL> DSA NAMING CONTEXT "/MTS=routing-domain-name"
  ```

  If the DSA indicates that no object exists, the DSA does not have knowledge of the naming context for the routing domain. Contact the person responsible for managing the directory to find out what the problem is.

- That you have used the correct password to authorize this system.

  It is possible that the password has been modified since you last entered information in the directory using the MTS entity.

9.3.6 Start the MTA

To start the MTA, execute the following procedure:

```
$ @SYS$STARTUP:MTA$COMMON_STARTUP
```

This MTA is now operational.

9.3.7 Check MTA$CREATE_CONS_TEMPLATES.NCL

If this MTA can connect to peer MTAs either within its routing domain, or in other routing domains using CONS, you need to check that the network described by the X.25 Template, referenced by the OSI Transport Template, is accessible to this MTA. If it is not, update the MTA's MTA$CREATE_CONS_TEMPLATES.NCL script to provide an alternative X.25 Template.

To change the X.25 Template referenced by the MTA, you must edit MTA$CREATE_CONS_TEMPLATES.NCL in SYS$SPECIFIC:[SYS$STARTUP] for this MTA and identify an alternative X.25 Template. This X.25 Template must describe an X.25 network that this MTA can directly communicate with. If you do not have a suitable X.25 Template defined on your system, the commands to create one are included in the MTA$CREATE_CONS_TEMPLATES.NCL script. Follow the instructions specified in the file to create a suitable template.

9.3.8 Updating System Startup Files

To make sure that the MTA startup script is executed when the system on which the MTA is set up is rebooted, add one of the following commands to the SYSTEM startup file for the node. In a cluster, make sure that the command is executed only by those nodes where an MTA is running:

```
$ @SYS$STARTUP:MTA$COMMON_STARTUP
```

```
$ SUBMIT SYS$STARTUP:MTA$COMMON_STARTUP /QUEUE = queue /NOPRINT
```

where *queue* is the name of a node-specific batch queue. In a cluster, you need to ensure that *queue* is served by the node that is currently rebooting.

9.3.9 What to Do Next

You have now completed the upgrade tasks for the MTA. You can continue at Section 9.6, which describes how to run the MTA verification procedure.

9.4 Setting Up an MTA in the Routing Domain for the First Time

This section describes the tasks that you need to complete when setting up MTAs for the first time within this routing domain. If you are upgrading from a previous release of the MTA, you do not need to read this section and should refer to Section 9.3.

If you have already set up one MTA and want to continue and set up more MTAs within the routing domain, refer to Section 9.5.

9.4.1 Set Up Access to a DSA

This section describes what you need to do to set up access to the DSA, namely:

- Configuring the DSA.
- Create a Naming Context, and possibly a Subordinate Reference entity, for the MTA's routing information at the DSA.
- Create a DUA defaults file.

9.4.1.1 Configure the DSA

On the system where you have installed the DSA, complete the following:

- Execute the following script:

  ```
  $ @SYS$STARTUP:DXD$DSA_CONFIGURE
  ```

  If a DSA is already created and running on this node, an informational message is displayed stating that a DSA already exists.

- Start the DSA as follows:

```
NCL> CREATE DSA
NCL> ENABLE DSA
```

If NCL returns errors for any of these commands, refer to the HP Enterprise Directory Service documentation for information on how to solve the problem.

9.4.1.2 Create a Naming Context for the MTA

Chapter 6 describes the MTA's directory requirements, in particular, whether a Subordinate Reference entity is required. If, after reading Chapter 6, you decide that you need a Subordinate Reference entity, refer to the HP Enterprise Directory Service documentation, which describes Subordinate Reference entities and how to create them in detail.

Note that you require write-access to the directory in order to create any entries in the directory.

The Subordinate Reference entity must exist at the system that holds the master copy of the Naming Context superior to the Naming Context that you want to create for your routing information.

Create a Naming Context entity for your routing domain on the DSA that will hold the master copy of the MTA's routing information as follows:

```
NCL> CREATE DSA NAMING CONTEXT /MTS=routing-domain-name
```

The DSA where you create the Naming Context entity is referred to as the **master DSA** for your routing domain.

If you receive an error indicating there is no parent entry for the Naming Context entity you are trying to create, there is a superior Naming Context defined that you are unaware of. Contact the person responsible for managing the directory, or refer to *HP Enterprise Directory - Problem Solving* for information about Naming Contexts.

9.4.1.3 Create the DUA Defaults File

Complete the following to create a DUA defaults file on the node where you intend to set up this MTA:

1. Check that the DSA is available as follows:

   ```
   NCL> SHOW NODE "dsa_node" DSA STATE
   ```

   where ***dsa_node*** is the name of the system where the master DSA for the routing information is installed.

   The DSA state must be ON. If the DSA is not ON, contact the person responsible for managing the DSA to find out whether there is a problem. You will not be able to continue until the DSA is ON.

2. Create a DUA defaults file.

   The MTA uses the HP Enterprise Directory Service DUA defaults file.

   Execute DXD$DUA_CONFIGURE as follows, and when prompted provide the name of the system where the master DSA is located:

   ```
   $ @SYS$STARTUP:DXD$DUA_CONFIGURE
   ```

   DXD$DUA_CONFIGURE creates the file DXD$DUA_DEFAULTS.DAT, which contains the Presentation address of the DSA. This address is then used by all the directory applications on the node to access the DSA.

If DXD$DUA_CONFIGURE fails to contact the master DSA, or fails to obtain information from the DSA, it displays an error message, and asks whether you want to try a different DSA. Answer **No**, because at this point you must use the master DSA where the Naming Context entity for the routing domain is created.

If the utility continues to fail, refer to *HP Enterprise Directory - Problem Solving* for information about problems using the DUA configuration utility.

9.4.2 Create the Routing Domain and Area Entries for the MTA

This section describes how to create the MTA's routing domain entry in the directory.

Creating the routing domain entry comprises:

- Creating the MTS entity and setting the MTS entity password
- Adding global domain identifiers
- Adding Area entities where required

You cannot create these entries unless a Naming Context entity already exists (see Section 9.4.1.2).

9.4.2.1 Create the MTS Entity for Your Routing Domain

This section describes how to create the MTS entity that represents your routing domain, and how to set the MTS entity password.

Before creating the MTS entity, you must initialize the MTS process required by the MAILbus 400 MTA as follows:

```
$ @SYS$STARTUP:MTA$MTS_INIT
```

You only need initialize the MTS process in this way the first time that you set up this MTA. At all other times, the MTS process is initialized within MTA$COMMON_STARTUP.COM.

If you do not have write-access to the directory, you will also need to quote values for the Authorizing User and Authorizing Password arguments when you create the MTS entity. However, if you have installed the directory solely for the MTA, you will not need authority as the directory is installed with write-access for all users. In this case, you can omit the Authorizing User and Authorizing Password arguments.

Create the MTS entity as follows:

```
NCL> CREATE MTS "/MTS=routing-domain-name" -
_NCL> PASSWORD "mts-password", -
_NCL> AUTHORIZING USER "directory-user", -
_NCL> AUTHORIZING PASSWORD "directory-pass"
```

where:

- ***mts-password*** is the password that you want to assign to your routing domain.

  Chapter 7 describes why a password is required.

- ***directory-user*** is the distinguished name of a user who has write-access to the directory.

- ***directory-pass*** is the Password attribute of the user with distinguished name ***directory-user***.

If NCL returns an error when you enter this command, check that you have used the correct syntax for the command. The *MTS Module Online Help* describes the syntax for the Create MTS command in detail.

You cannot create these entries unless a Naming Context entity already exists (see Section 9.4.1.2).

Do not forget the password that you enter when you create the MTS entity as you must later provide it on all other systems where you intend to manage routing information. Later sections in this chapter describe how you can use this password to ensure that you can manage the MTA routing information from other systems in the routing domain.

9.4.2.2 Add Global Domain Identifiers

Add the list of GDIs for your routing domain to the routing domain entry:

```
NCL> SET MTS "/MTS=routing-domain-name" -
_NCL> GLOBAL DOMAIN IDENTIFIERS -
_NCL> {"C=c-name;A=admd-name;P=prmd-name",...}
```

where:

- ***c-name*** is the Country Name attribute for the GDI.

- ***admd-name*** is the Administration Domain Name attribute for the GDI.
- ***prmd-name*** is the Private Domain Name attribute for the GDI.

9.4.2.3 Create Area Entries

If you have not planned any areas for your routing domain, continue at Section 9.4.3.

If you have planned to divide your routing domain into areas, create Area entities for each area in your routing domain:

```
NCL> CREATE MTS "/MTS=routing-domain-name" AREA "area-name"
```

where ***area-name*** is an area name.

If you have planned a hierarchy of areas, for example, area1.area2, create the most significant area in the hierarchy first. This ensures that the area hierarchy is stored correctly in the directory. Create the Area entities for area1 and area2, where area2 is subordinate to area1, as follows:

```
NCL> CREATE MTS "/MTS=routing-domain-name" AREA "area1"
NCL> CREATE MTS "/MTS=routing-domain-name" AREA "area1.area2"
```

9.4.3 What to Do Next

You have completed setting up the MTA routing domain entry in the directory. You can continue and run the MTA Setup procedure as described in Section 9.5.3.

9.5 Adding More MTAs to Your Routing Domain

This section assumes that one MTA has been set up and is operational. Complete this section for each additional MTA that you want to operate within your routing domain on another system.

If you are upgrading an MTA, refer to Section 9.3.

If you want to manage your routing information from a system where you are not operating an MTA Server, you can set up the management component of the MTA without the Server (see Section 9.10).

The DSA that an MTA will connect to will already be installed and set up. This DSA might be on the system where you intend to set up the MTA, or it might be on a different system. If the DSA is on a different system, you need to know the name of the system where it is running.

In the following subsections you will:

- Set up the MTA's access to the master DSA.

  When you have completed setting up the MTA and confirmed that the set up is successful, you can then choose to modify the MTA setup such that the MTA contacts a shadow DSA. This is described in Section 9.8.

- Run the MTA Setup procedure.
- Create MTA entries in the directory.

9.5.1 Set Up Access to a DSA

On the system where you have installed the MTA components, complete the following:

1. Check that the DSA is available as follows:

   ```
   NCL> SHOW NODE "dsa_node" DSA STATE
   ```

   where ***dsa_node*** is the name of the system where the master DSA is installed.

   The DSA state must be ON. If the DSA is not ON, contact the person responsible for managing the DSA to find out whether there is a problem. You will not be able to continue until the DSA is ON.

2. Create a DUA defaults file.

   Execute DXD$DUA_CONFIGURE as follows, and when prompted provide the name of the system where the master DSA to which you intend this MTA to connect is located:

```
$ @SYS$STARTUP:DXD$DUA_CONFIGURE
```

DXD$DUA_CONFIGURE creates the file DXD$DUA_DEFAULTS.DAT, which contains the Presentation address of the DSA. This address is then used by all the directory applications on the node to access the DSA.

If DXD$DUA_CONFIGURE fails to contact the master DSA, or fails to obtain information from the DSA, it displays an error message, and asks whether you want to try a different DSA. Answer **No** because at this point you must use the master DSA where the Naming Context entity for the routing domain is created.

If the utility continues to fail, refer to *HP Enterprise Directory - Problem Solving* for information about problems using the DUA configuration utility.

3. Initialize the MTS process required by the MAILbus 400 MTA:

```
$ @SYS$STARTUP:MTA$MTS_INIT
```

You only need initialize the MTS process in this way the first time that you set up this MTA. At all other times, the MTS process is initialized within MTA$COMMON_STARTUP.COM.

4. Check that the DSA has knowledge of the naming context for the routing domain as follows:

```
NCL> SHOW NODE "dsa_node" -
_NCL> DSA NAMING CONTEXT "/MTS=routing-domain-name"
```

If you have set up your first MTA using the instructions in *MAILbus 400 Getting Started*, *routing-domain-name* is the same as *email-system-name* used in *MAILbus 400 Getting Started*.

If the DSA indicates that no such object exists, the DSA does not have knowledge of the naming context for the routing domain. In this case, contact the person responsible for managing the directory to find out what the problem is.

5. If the DSA does have knowledge of the naming context, check that the DSA also has knowledge of the routing information for your routing domain as follows:

```
NCL> SHOW MTS "/MTS=routing-domain-name"
```

If the DSA does not have knowledge of the MTA's routing information, check that the DXD Transport Templates have been set up correctly. The HP Enterprise Directory Service documentation explains how to set up the DXD Transport Templates.

When you have confirmed that the DSA has knowledge of your routing information, you can continue to set up the MTA.

9.5.2 Provide Authorization to Manage the MTA's Routing Information

Provide the necessary authorization to enable you to manage the MTA's routing information from this system. Use the following command to authorize the MTS entity:

```
NCL> AUTHORIZE MTS "/MTS=routing-domain-name" PASSWORD "mts-password"
```

where ***mts-password*** is the password that you provided when you created the routing domain entry (see Section 9.4.2.1).

Continue at Section 9.5.3 and run the MTA Setup procedure.

9.5.3 Run the MTA Setup Procedure

This section describes how to use the MTA Setup procedure to set up an MTA.

Complete the following steps to set up an MTA on the node where you want the MTA to run:

1. Execute the following procedure:

   ```
   $ @SYS$STARTUP:MTA$SERVER_SETUP
   ```

2. The procedure asks a series of questions. If you have executed the setup procedure before, your previous settings are offered as defaults. You can request Help for any question by responding to the question with a question mark (?):

   a. `Do you want to continue with this procedure? [yes]`

      If, after reading the overview of what the MTA setup procedure will do, you decide not to continue, enter **No** and exit from the MTA setup procedure; otherwise, enter **Yes**.

Note

Questions b, c, and d are only asked the first time you set up an MTA. When you subsequently run MTA$SERVER_SETUP.COM, including following an upgrade of the product, these questions are omitted.

This means that you cannot change the device, or devices, where the MTA's Workspace directories, Accounting directory or Archive directories are located once you have specified them using this setup procedure. *HP MAILbus 400 MTA Tuning and Problem Solving* describes Accounting and Archiving.

   b. `Enter the device for the MTA's Workspace directories`

Supply the name of the device where you want the MTA's Workspace directories to be created. You can specify a physical device name or a logical device name. You must not use a rooted logical to name the device.

If you are setting up the MTA on a cluster, you can use the same device to hold more than one MTA's Workspace directory. The MTA's Workspace directories contain the name of the node and are therefore unique.

c. `Enter the device for the MTA's Accounting directory`

Supply the name of the device where you want the MTA's Accounting directory to be created. You can specify a physical device name or a logical device name. You cannot use a rooted logical to name the device. The name of the device specified for the MTA's Workspace directory is offered as a default.

If you have more than one device available to the node where you want to set up this MTA, you are advised to use an alternative device for the MTA's Accounting directory.

If you are setting up the MTA on a cluster, you can use the same device to hold more than one MTA's Accounting directory. The MTA's Accounting directory name contains the name of the node and is therefore unique.

d. `Enter the device for the MTA's Archive directories`

Supply the name of the device where you want the MTA's Archive directories to be created. You can specify a physical device name or a logical device name. You cannot use a rooted logical to name the device. The name of the device specified for the MTA's Accounting directory is offered as a default.

If you have more than one device available to the node where you want to set up this MTA, you are advised to use an alternative device for the MTA's Archive directory.

If you are setting up the MTA on a cluster, you can use the same device to hold more than one MTA's Archive directories. The MTA's Archive directories contain the name of the node and are therefore unique.

e. `What is the distinguished name of your routing domain?`

Supply the distinguished name of the routing domain. This is the name that you used to create the MTS entity (the routing domain entry). See Section 9.4.2.1.

f. What is the name of the MTA?

Supply the name of the MTA, including any area identifiers. The area identifiers are those specified in Section 9.4.2.3.

The MTA name must be unique within the routing domain and be in the form: area.mta_name.

g. What is the MTA password?

Supply a password for the MTA. The password can be from 1 to 62 characters from the printable string character set. Note that you must not enter a null password.

The MTA setup procedure enters the command to create the MTA entity into the script SYS$SPECIFIC:[SYS$STARTUP]MTA$START.NCL. The MTA setup procedure also creates the file MTA$MTS_CREATE_MTA_ENTRY.NCL in SYS$SPECIFIC:[SYS$STARTUP]. This file contains the NCL commands to create the entry for the MTA in the directory.

The MTA Setup procedure includes all the NSAPs that it can detect on the system where the setup procedure is executed in the MTA's Presentation and Session addresses, which is located at SYS$SPECIFIC:[SYS$STARTUP]MTA$MTS_CREATE_MTA_ENTRY.NCL. However, in the case of RFC 1006, if the MTA cannot determine the TCP/IP address, you will need to add it manually to MTA$MTS_CREATE_MTA_ENTRY.NCL in SYS$SPECIFIC:[SYS$STARTUP].

The MTA will only be able to determine the TCP/IP address automatically where the HP TCP/IP Service for OpenVMS is installed. If the MTA cannot find a TCP/IP address, you will see the following informational message:

```
Cannot determine the RFC1006 NSAP of this system

If you require the MTA to operate over RFC 1006 then you must manually
edit the file SYS$SPECIFIC:[SYS$STARTUP]MTA$MTS_CREATE_MTA_ENTRY.NCL
and add the RFC 1006 NSAP to the MTA's Presentation and Session
addresses.

The syntax for an RFC1006 NSAP is:
 "RFC1006+<IP address or hostname>,RFC1006" for example
 "RFC1006+1.2.3.4,RFC1006"
```

The following is an example of a CLNS and RFC 1006 NSAP within an MTA's Presentation address:

```
"""MTA""/""MTA""/""MTA88-NODE1""/NS+4900AB01002CD03E45521,CLNS|
  RFC1006+1.2.3.4+102,RFC1006"
```

In this example, “102” is the TCP port used by the MAILbus 400 MTA. *HP MAILbus 400 MTA Tuning and Problem Solving* provides more information about TCP/IP addresses when used by the MTA.

9.5.4 Create the MTA Entry in the Directory

When you are satisfied that all the required NSAPs are included in the MTA’s Presentation and Session addresses, create the MTA entry in the directory. Execute MTA$MTS_CREATE_MTA_ENTRY.NCL, which creates an MTA entity as a subentity of the MTS entity as follows:

```
NCL> DO SYS$STARTUP:MTA$MTS_CREATE_MTA_ENTRY.NCL
```

If this command fails, check:

- That you have created the appropriate MTS and Area entities in the directory (see Section 9.4.2).
- That the routing information is accessible to the master DSA as follows:

  ```
  NCL> SHOW NODE "dsa_node" -
  _NCL> DSA NAMING CONTEXT "/MTS=routing-domain-name"
  ```

  If the DSA indicates that no object exists, the DSA does not have knowledge of the naming context for the routing domain. Contact the person responsible for managing the directory to find out what the problem is.
- That you have used the correct password to authorize this system.

  It is possible that the password has been modified since you last entered information in the directory using the MTS entity.

9.5.5 Start the MTA

To start the MTA, execute the following procedure:

```
$ @SYS$STARTUP:MTA$COMMON_STARTUP
```

This MTA is now operational.

If you intend to communicate with MTAs using CONS, you need to complete the tasks described in Section 9.5.6. If you do not intend to use CONS, you can continue at Section 9.5.7.

9.5.6 Check MTA$CREATE_CONS_TEMPLATES.NCL

If this MTA can connect to peer MTAs either within its routing domain, or in other routing domains using CONS, you need to check that the network described by the X.25 Template, referenced by the OSI Transport template, is accessible to this MTA. If it is not, update the MTA's MTA$CREATE_CONS_TEMPLATES.NCL script to provide an alternative X.25 Template.

To change the X.25 Template referenced by the MTA, you must edit MTA$CREATE_CONS_TEMPLATES.NCL in SYS$SPECIFIC:[SYS$STARTUP] for this MTA and identify an alternative X.25 Template. This X.25 Template must describe an X.25 network with which this MTA can directly communicate. If you do not have a suitable X.25 Template defined on your system, the commands to create one are included in the MTA$CREATE_CONS_TEMPLATES.NCL script. Follow the instructions specified in the file to create a suitable template.

9.5.7 Updating System Startup Files

To make sure that the MTA startup script is executed when the system on which the MTA is set up is rebooted, ensure that the following command to execute the MTA startup procedure is added to the System startup file for the node, or cluster:

```
$ SUBMIT SYS$STARTUP:MTA$COMMON_STARTUP /QUEUE = queue /NOPRINT
```

where *queue* is the name of a node-specific batch queue. In a cluster, you need to ensure that *queue* is served by the node that is currently rebooting.

9.6 Using the MTA Verification Procedure

Use the MTA verification procedure to check that your MTA installation and setup has been successful.

To run the MTA verification procedure, enter the following command from the SYSTEM account on the node where you have set up the MTA:

```
$ @SYS$TEST:MTA$VP
```

If the MTA verification procedure fails for any reason, error messages are logged in SYS$TEST:MTA$VP.LOG

You can run the verification procedure at any time, provided the MTA is in the ON state, and the MTA is contacting the master DSA.

Note

If the MTA is contacting a shadow DSA, that is, a DSA that holds a copy of the MTA routing information, the procedure will fail.

The verification procedure confirms that a particular MTA is running correctly. The verification procedure confirms that:

- A message can be submitted from an Agent to the MTA.
- A message can be delivered back to the Agent from the MTA.
- The MTA can access the directory.

The following are examples of problems that the verification procedure can report:

- The MTA is not created and enabled.
- The verification procedure is not authorized to access the routing information from the node.
- The DSA is not available.

9.7 Keeping the MTA Startup Script Up-to-Date

You need to execute the MTA startup procedure every time you want to start the MTA. Changes made interactively to the MTA through NCL management are not automatically written to the MTA startup script SYS$SPECIFIC:[SYS$STARTUP]MTA$START.NCL and will be lost each time the MTA is shut down. For this reason, it is important that you regularly update the MTA startup script and any other Peer MTA or Agent scripts to

include any changes to the attributes of the MTA or its subentities that you make through NCL management.

9.8 Setting Up an MTA to Contact a Shadow DSA

This section describes how to set up your MTA to contact a shadow DSA. You must first make sure that you have set up the MTA as described in the earlier sections in this chapter.

Before you modify the MTA setup, make sure that you have replicated all the MTA's routing information to the shadow DSA (see the HP Enterprise Directory Service documentation).

On the system where the MTA that you want to modify is set up, complete the following:

1. Check that the DSA that you intend the MTA to contact is available as follows:

   ```
   NCL> SHOW NODE "dsa_node" DSA STATE
   ```

   where ***dsa_node*** is the name of the system where the DSA is installed.

   The DSA state must be ON. If the DSA is not ON, contact the person responsible for managing the DSA to find out whether there is a problem. You will not be able to continue until the DSA is ON.

2. Shutdown the MTA as follows:

   ```
   $ @SYS$SYSTEM:[SYS$STARTUP]MTA$COMMON_SHUTDOWN
   ```

3. Create a new DUA defaults file.

   Execute DXD$DUA_CONFIGURE as follows and, when prompted, provide the name of the system where the DSA to which you intend the MTA to connect is located:

   ```
   $ @SYS$STARTUP:DXD$DUA_CONFIGURE
   ```

   Executing this command will update the file DXD$DUA_DEFAULTS.DAT, which contains the Presentation address of the DSA. This address is then used by all the directory applications on the node to access the DSA.

   If DXD$DUA_CONFIGURE fails to contact the DSA on the specified node, or fails to obtain information from the DSA, it displays an error message, and asks whether you want to try a different DSA. Answer **No** and continue at step 6, and the MTA will continue to use the routing information at the master DSA.

4. Check that the DSA has knowledge of the naming context for the routing domain as follows:

   ```
   NCL> SHOW NODE "dsa_node" -
   _NCL> DSA NAMING CONTEXT "/MTS=routing-domain-name"
   ```

   If the DSA indicates that no such object exists, the DSA does not have knowledge of the naming context for the routing domain. In this case, contact the person responsible for managing the directory to find out what the problem is.

5. If the DSA does have knowledge of the naming context, check that the DSA also has knowledge of the routing information for your routing domain as follows:

   ```
   NCL> SHOW MTS "/MTS=routing-domain-name"
   ```

   If the DSA does not have knowledge of the MTA's routing information, check that the MTA's routing information has been successfully replicated to the shadow DSA. In particular, check that the master and shadow DSAs "trust" each other. Refer to the HP Enterprise Directory Service documentation for details of replication and trust agreements.

   When you have confirmed that the DSA has knowledge of your routing information, you have completed the task of setting up your MTA to contact a shadow DSA. Make sure that you update the shadow DSA whenever you modify the MTA routing information in the directory.

6. Startup the MTA as follows:

   ```
   $ @SYS$SYSTEM:[SYS$STARTUP]MTA$COMMON_STARTUP
   ```

9.9 What to Do Next

You have now completed setting up an MTA within your routing domain. You can now continue and choose to:

- Upgrade more MTAs in your routing domain as described in Section 9.2 and Section 9.3.
- Set up more MTAs in your routing domain as described in Section 9.2 and Section 9.5.
- Set up the routing information for your routing domain:
  - Store the remainder of your routing information in the directory.

    This is information such as O/R addresses and information relating to other routing domains (see Section 10.1).

– Create entities that represent peer MTAs locally at the MTA.

  These are Peer MTA entities and are described in Section 10.2.

– Create entities that represent Agents locally at the MTA.

  These are Agent entities and are described in Section 10.3.

9.10 Setting Up Remote Access to the Routing Information in the Directory

This section describes how you set up a system in order to manage the MTA routing information in the directory, without setting up an MTA.

Before you set up remote access to your routing information on a node or cluster, make sure that you have the following components installed:

- HP Enterprise Directory Service Base

  Note that, for a cluster, the device for the [DXD$SERVER] directory must be available to those nodes in the cluster where you want to be able to manage the MTA routing information remotely.

- MAILbus 400 MTA Mgt

Complete the following steps on the node, or cluster, where you have installed the required components:

1. Create a DUA defaults file:

   The MTA uses the directory DUA defaults file. If there is already a directory application installed on this node, there will be a DUA defaults file located at DXD$DIRECTORY:DXD$DUA_DEFAULTS.DAT.

   If there is a DUA defaults file already set up on the node, continue at step 2.

   If no defaults file exists on the node, execute DXD$DUA_CONFIGURE and when prompted provide the name of the node where the DSA to which you want the node to connect is located:

   ```
   $ @SYS$STARTUP:DXD$DUA_CONFIGURE
   ```

   DXD$DUA_CONFIGURE creates the file DXD$DUA_DEFAULTS.DAT, which contains the Presentation address of the DSA. This address is then used by all the directory applications on the node.

   Note that you only need to run DXD$DUA_CONFIGURE once in a cluster.

If DXD$DUA_CONFIGURE fails to contact the DSA on the specified node, or fails to obtain information from the DSA, it displays an error message, and asks whether you want to try a different DSA. You can name an alternative DSA that you know holds a copy of the routing information for your routing domain.

If the utility continues to fail, refer to the *HP Enterprise Directory - Problem Solving* for information about problems using the DUA configuration utility.

2. Initialize the MTS process:

```
$ @SYS$STARTUP:MTA$MTS_INIT
```

You only need to initialize the MTS process in this way the first time that you set up this MTA. At all other times, the MTS process is started within MTA$COMMON_STARTUP.COM.

3. Check that the DSA has knowledge of the naming context for the routing domain as follows:

```
NCL> SHOW NODE "dsa_node" -
_NCL> DSA NAMING CONTEXT  "/MTS=routing-domain-name"
```

If the DSA indicates that no such object exists, the DSA does not have knowledge of the naming context for the routing domain. In this case, contact the person responsible for managing the directory to find out what the problem is.

4. Provide the necessary authorization so you can manage the MTA's routing information from this node.

Note that if this is an upgrade from a previous version of the MTA, the system will already be authorized. The exception to this is if the password has been modified during the upgrade.

```
NCL> AUTHORIZE MTS "/MTS=routing-domain-name PASSWORD "password"
```

where ***password*** is the password that you provided when you created the routing domain entry.

10

Setting Up the Remaining Routing Information

If you are upgrading from a previous version of the MAILbus 400 MTA and you have already set up all your routing information, you do not need to read this chapter.

Routing information stored in the directory is entered once using the entities of the MTS module. This information is described in Section 10.1. Chapter 7 gives an example, based on the ACME Shoe Corporation, of entering routing information in the directory. The *MTS Module Online Help* describes the commands used to create routing information in the directory.

Routing information held locally at MTAs is held in entities of the MTA module. This routing information must be added to the MTA startup script because any modifications made interactively are not preserved after the MTA is shut down. This information includes that described in Section 10.2 and Section 10.3. The *MTA Module Online Help* describes the commands to create the Agent and Peer MTA entities.

10.1 Routing Information Stored in the Directory

There is a Load Countries script provided with the MTA. This script contains the NCL commands to add the country codes as defined in ISO International Standard 3166, and as listed in Appendix D. The script is located at:

Tru64 UNIX

/var/mts/scripts/populate_countries.ncl

♦

OpenVMS

SYS$SPECIFIC:[SYS$STARTUP]MTA$MTS_POPULATE_COUNTRIES.NCL

♦

You can use the script if you need to specify routing information for individual countries for your routing domain. Follow the instructions provided in the script.

The following list briefly describes the tasks that you need to complete to set up your required routing information:

- Create entries for any MTA sets in the directory. You do not need to create entries for the members of the MTA set as this is done when the individual MTAs are installed and set up.
- Set up the information required for individuals in your routing domain.

  Create O/R addresses in the directory for the individuals served by User Agents and add the routing instruction(s) for these O/R addresses. You will also need to create information to be held at the MTA about each registered User Agent that the individuals use. This information is held in an Agent entity, at the MTA that serves the registered User Agent (see Section 10.3).
- Set up the information required for the individuals with whom you want to communicate in other X.400 routing domains.

  This includes:

  – Creating partial O/R address entries in the directory for the user community in each routing domain and adding the routing instruction(s) for these O/R addresses.

  – Creating domain entries in the directory for the other X.400 routing domains and adding the routing instructions for the routing domains to the entries.

  You will also need to create information to be held at the MTA about the Peer MTA that serves the individuals. This information is held in a Peer MTA entity, at the boundary MTA that connects to the peer MTAs in the other routing domains (see Section 10.2).
- Set up the information required for the individuals with whom you want to communicate in other routing domains to which your MTAs connect through a Gateway.

  This includes:

  – Creating an O/R address entry in the directory. This is either a partial O/R address entry, or complete O/R address entries, for the users in the routing domain served by the Gateway.

  – Creating an entry in the directory for the routing domain served by the Gateway and adding the routing instruction for the routing domain to the entry.

- If the Gateway supports address translation, creating foreign address entries in the directory for the users in your routing domain and the routing domain served by the Gateway.

You will also need to create information to be held at the MTA about the Gateway that serves the individuals. This information is held in an Agent entity, at the MTA that serves the Gateway (see Section 10.3).

You might also need to refer to the documentation supplied with the Gateway for details of the routing information to add to these entries.

Appendix B provides an example of some of the commands used to set up routing information in the directory for the ACME Shoe Corporation. This script is also provided online at:

Tru64 UNIX

/var/mts/scripts/populate_mts_example.ncl

♦

OpenVMS

SYS$SPECIFIC:[SYS$STARTUP]MTA$MTS_POPULATE_EXAMPLE.NCL

♦

HP MAILbus 400 MTA Tuning and Problem Solving provides some examples of how the MTA uses routing information to route messages.

10.2 Adding Peer MTA Entities at the MTA

This section describes how to add Peer MTA entities at an MTA that you have already set up.

If the MTA that you have set up is a boundary MTA to one or more peer MTAs in other routing domains, you need to create Peer MTA entities for each peer MTA to which the boundary MTA connects. Use the information planned in Chapter 4 and listed in Table 4–2.

When you are sure that the peer MTA connections are correctly set up, you can add the management commands for the respective entities to the MTA Startup script.

If you want to add modifications to the MTA Startup script, you are advised to create your own NCL scripts and add the command to execute the script to the startup script, for example:

```
do mta-peer-mta-script.ncl
```

where *mta-peer-mta-script.ncl* is the complete filename of a script that defines one or more peer MTAs. Edit the Peer MTA scripts with details of the peer MTAs to which the MTA can connect.

In this way, if the MTA Startup script is modified between versions, it is easier for you to add your customizations to the new file.

The MTA Startup script contains example commands for creating Peer MTA entities. The examples include a Peer MTA entity representing a peer MTA in a routing domain based on the 1984 MHS standards, for example, the VAX Message Router X.400 Gateway.

A full list of the characteristic attributes of the Peer MTA entity can be found in the *MTA Module Online Help* and *HP MAILbus 400 MTA Tuning and Problem Solving*.

10.3 Adding Agent Entities at the MTA

This section describes the tasks that you need to complete to create Agent entities for each Agent with which the MTA will communicate. Use the information planned in Chapter 4 and listed in Table 4–3.

When you are sure that the Agent connections are correctly set up, you can add the management commands for the respective entities to the MTA Startup script.

If you want to add modifications to the MTA Startup script for Agents that you create, you are advised to create your own NCL scripts and add the command to execute the script to the MTA Startup script, for example:

```
do mta-agent-script.ncl
```

where *mta-agent-script.ncl* is the complete name of a script that defines one or more Agents.

In this way, if the MTA Startup script is modified between versions, it is easier for you to add your Agent customizations to the new file.

The MTA Startup script contains example commands for creating Agent entities. The examples include an Agent, which uses the Shared File 1984 or 1992 interface, and an XAPI Agent. Edit the Agent startup scripts and the MTA script with details of the Agent or Agents that the MTA serves.

A full list of the characteristic attributes of the Agent entity can be found in the *MTA Module Online Help* and *HP MAILbus 400 MTA Tuning and Problem Solving*.

Chapter 11 describes how to make sure that an Agent can connect to the MTA.

11

Setting Up Connections from Agents to the MTA

This chapter describes how to make sure Agents can connect to the MAILbus 400 MTA.

The chapter assumes that you have created an Agent entity, an entity of the MTA module, as described in Section 10.3 and that you have followed all relevant instructions in the Agent documentation. The chapter also assumes that the Agent is installed.

Note that this chapter describes tasks that might also be described in the Agent documentation. You are advised to read the information included in this chapter, in addition to the Agent documentation, to ensure that all the tasks necessary to set up connection between the Agent and the MTA are complete.

The tasks required to set up an Agent for use with the MTA vary depending on the type of interface the Agent uses. Section 11.1 describes the setup for XAPI Agents and Section 11.2 describes the setup for Shared File Agents.

11.1 Setup for Agents That Use the XAPI Interface

This section describes the information that is automatically set up for XAPI Agents by the MAILbus 400 MTA when the MTA is installed.

Agents installed on the same system as the MTA can connect to the MTA either directly or using the API Server. Agents installed on a system remote from the MTA can only connect to the MTA through the API Server.

If you do not know whether the Agent uses the API Server, assume it does not and do not make any modifications as described in Section 11.1.1. If the MTA and the Agent can communicate, no more setup is required. However, if the Agent and the MTA cannot communicate, make the changes described in Section 11.1.2.

Examples of Agents that use the XAPI interface are:

- HP MAILbus 400 SMTP Gateway

- MAILbus 400 Message Store
- LinkWorks
- MailWorks Server for Tru64 UNIX

11.1.1 Agents That Connect Directly to the MTA

When the MTA is installed, it automatically:

Tru64 UNIX

Creates a file that specifies the address of the API Server. This is the address of the system where the MTA is installed. The file is /var/mta/mta_api_server_address and contains the string `LOCAL`.

♦

OpenVMS

Creates the startup file MTA$CLIENT_STARTUP.COM in the SYS$COMMON:[SYS$STARTUP] directory on the system where the MTA is installed. The MTA automatically executes this file when it is started. The startup file:

- Defines the logical name MTA_NODE to be the node where the MTA is installed.
- Invokes MTA$CREATE_CLIENT_CLNS_TEMPLATES.NCL, which creates the template for the connection to the MTA.

The logical MTA_NODE can be overridden for individual processes by using a process logical name. The exact form of the logical is:

`LOCAL:MTA_CLIENT_CLNS%`*`nsap_address`*

where *`nsap_address`* is the NSAP address of the node where the MTA is installed.

♦

In most cases (that is, where the Agent is located on the same system as the MTA) no more action is required. However, for Agents that use the XAPI interface to connect to the API Server, you will need to set up the Agent as described in Section 11.1.2.

11.1.2 Agents That Use the API Server

Agents that use the XAPI interface might be installed on either the same system as the MTA, or on a system that is remote from the MTA.

Agents that are installed on a system remote from the MTA require the MAILbus 400 MTA Base to be installed on the Agent system before the Agent can be set up. Refer to the MTA installation documentation for instructions on how to install MAILbus 400 MTA Base.

When the Agent connects to the MTA through the API Server, complete the following:

Tru64 UNIX

Change the string LOCAL in the file /var/mta/mta_api_server_address to one of the following, depending on the transport type on the system where the Agent is installed:

- For connections using OSI transport (TP4), change the first line of mta_api_server_address to:

  OSITP *mta_nsap_address*

  where *mta_nsap_address* is the NSAP address of the node where the MTA that serves the Agent is running. Note that the MTA may be running on either the same node as the Agent or on a different node.

  For example:

  OSITP 49002aaa00040066aa21

- For connections using TCP/IP change the first line of mta_api-server_address to:

  TCP *hostname*

  where *hostname* can be one of:

  – A hostname, for example, “apple”

  – A hostname, plus subdomain and domain names, for example, “apple.sub.sub.dom”

  – An IP address, for example, 1.2.3.4

  Note that this can be either the same node as the node on which the Agent is running, or it can be a different node.

Also, for an Agent that connects to the MTA over TCP/IP, if other applications use the TCP port number 200, you will need to change the port number that the Agent uses to connect to the MTA. Refer to *HP MAILbus 400 MTA Tuning and Problem Solving* for details of how to do this.

♦

OpenVMS

Chagne the definition of the logical MTA_NODE in MTA$CLIENT_STARTUP.COM to one of the following, depending on the transport type on the system where the Agent is installed:

- For connections using OSI Transport (TP4), define MTA_NODE as:

OSITP:MTA_CLIENT_CLNS%*mta_nsap_address*

where *mta_nsap_address* is the NSAP address of the node where the MTA that serves the Agent is running. Remember that the MTA may be running on the same node as the Agent or on a different node.

- For connections using NSP (DECnet transport) define MTA_NODE as:

  NSP:*mta_node_name*

  where *mta_node_name* is the name of the node where the MTA that serves the Agent is running. Note that this may be either the same node as the node on which the Agent is running, or a different node.

- For connections using TCP/IP, define MTA_NODE as:

  TCP:*hostname*

  where *hostname* can be one of:

  – A hostname, for example, “apple”

  – A hostname, plus subdomain and domain names, for example, “apple.sub.sub.dom”

  – An IP address, for example, 1.2.3.4

  You must use HP TCP/IP Services for OpenVMS when setting up TCP/IP Transport Service connections for the MAILbus 400 MTA.

If the Agent is running on a system that is remote from the MTA, make sure that the MTA$CLIENT_STARTUP.COM file on the Agent system is run every time the system is rebooted. For example, include it in the system startup file on the system where the Agent is installed.

♦

11.2 Setup for Gateways That Use the Shared File Interface

This section describes how to set up the MTA for Gateways that use the Shared File 1984 or 1992 interface. Examples of such Gateways are:

- Gateways developed for the ISOPLEX 800 MTA:

  – ISOGATE™ for cc:Mail®, which enables users of an electronic messaging system based on cc:Mail to connect to the MAILbus 400 environment.

- ISOGATE for MS Mail, which enables users of an electronic messaging system based on MS Mail to connect to the MAILbus 400 environment.
- ISOCOR™ X.400 Router for Lotus Notes®, which enables users of Lotus Notes to connect to the MAILbus 400 environment.

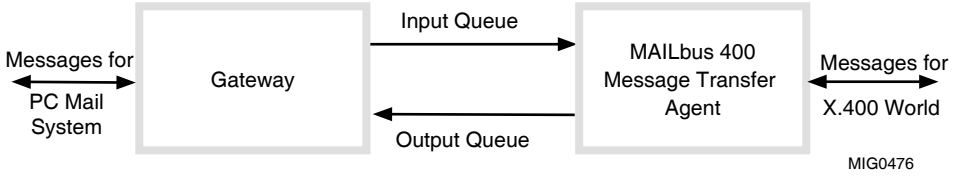
These Gateways can connect directly to the MAILbus 400 MTA. The MAILbus 400 MTA's Shared File 1992 interface provides the equivalent level of service to that provided by the ISOPLEX 800 MTA.

- PC LAN Gateways developed for the Retix OpenServer 400
  - OpenServer 400 cc:Mail Gateway to X.400, which enables cc:Mail users to connect to the MAILbus 400 environment.
  - OpenServer 400 Microsoft Mail for PC Networks Gateway to X.400, which enables Microsoft® Mail users, on a PC, to connect to the MAILbus 400 environment.
  - OpenServer 400 NetWare® MHS Gateway to X.400, which enables users of MHS-based mail systems to connect to the MAILbus 400 environment.

  These Gateways can connect directly to the MAILbus 400 MTA. The MAILbus 400 MTA's Shared File 1984 interface provides the equivalent level of service to that provided by the Retix OpenServer 400.

Figure 11–1 shows how the MAILbus 400 MTA and a Gateway using the Shared File interface exchange messages.

Figure 11–1 MTA Input and Output Queues



An MTA creates Input and Output queues, and a gateway.dat file for each Agent that uses the Shared File interface. Section 11.2.1 describes the MTA Input and Output queues and gateway.dat in more detail.

These queues and gateway.dat are created when the Agent entity representing the Agent is created. You create Agent entities for each of the Gateways that you want to use with the MTA in the same way as for other Agents, as part of MTA set up (see Chapter 8 (Tru64 UNIX) or Chapter 9 (OpenVMS)).

When using the MAILbus 400 MTA with these Gateways, you need a product that enables personal computers (PCs) to take advantage of the resources available on host computers, such as file services. The product needs to be installed on the PC where you want to use the Gateway and on the node where the MAILbus 400 MTA is installed. PATHWORKS and Network File Service (NFS) are examples of such products.

Make sure that you have set up the file services so that the MAILbus 400 MTA Input and Output queues are accessible to the PCs where the Gateway is installed.

Using this software, the Gateway can access messages held on the node where the MAILbus 400 MTA is installed. These messages are held in an MTA's Input and Output queues and are shared by the MTA and the PC.

11.2.1 MAILbus 400 MTA Queues

The following are the locations of the MTA Input and Output queues and gateway.dat. These Input and Output queues are implemented as directories, which are created by the MTA for each Gateway using the Shared File 1984 or 1992 interface:

Tru64 UNIX

/var/mta/workspace/agents/*agent_name*/iq

This is the Input queue used by the MAILbus 400 MTA for the Agent specified by *agent_name*.

The MTA imports messages from the Gateway on the Input queue for processing.

/var/mta/workspace/agents/*agent_name*/oq

This is the Output queue used by the MAILbus 400 MTA for the Agent specified by *agent_name*.

The MTA places messages for export to a Gateway on the Output queue.

/var/mta/workspace/agents/*agent_name*/gateway.dat

This file is used by the Gateway identified by *agent_name* to confirm the pathnames of the Input and Output queues.

Within management, Agent names can be specified in upper or lower case. However, they are converted to lower case when used in the pathname for the Input queue, Output queue, and gateway.dat.

♦

OpenVMS

device:[MTA$*node*.WORKSPACE.AGENTS.*agent_name*.IQ]

This is the Input queue used by the MAILbus 400 MTA for the Agent specified by *agent_name*.

The MTA imports messages from the Gateway on the Input queue for processing.

device:[MTA$*node*.WORKSPACE.AGENTS.*agent_name*.OQ]

This is the Output queue used by the MAILbus 400 MTA for the Agent specified by *agent_name*.

The MTA places messages for export to a Gateway on the Output queue.

device:[MTA$*node*.WORKSPACE.AGENTS.*agent_name*]GATEWAY.DAT

This file is used by the Gateway identified by *agent_name* to confirm the directories of the Input and Output queues.

The variables in the above locations mean the following:

- *device* is the device that you specified for the MTA's Workspace directory at setup.
- *node* is the systems communication services (SCS) name for the node where the MTA to which the Agent connects is running.
- *agent_name* is the name given to the Agent.

♦

11.2.2 Contents of gateway.dat for Shared File 1984 Interface

The following is an example of the file gateway.dat for Agents that use the Shared File 1984 interface:

```
E:/AGENT_NAME/IQ
E:/AGENT_NAME/OQ gateway_name <other parameters>
```

where:

- *AGENT_NAME* is the name that you have assigned to the Agent entity that represents the Gateway at the MAILbus 400 MTA that serves the Gateway.

  Note that the Agent name is converted to upper case when included in the contents of gateway.dat.

- *gateway_name* is the name of the Gateway as supplied when the Gateway is configured.

You must edit gateway.dat to include *gateway_name* and other parameters, as specified in the Agent documentation, after it is created by the MTA. For consistency you are advised to use the name of the Gateway's Agent entity as the gateway name.

The separator used in the directory specification is "/" and not the standard DOS separator "\".

Example 11–1 is an example of the contents of the file gateway.dat for the OpenServer 400 Netware MHS Gateway.

Example 11–1 Example gateway.dat for OpenServer 400 NetWare MHS Gateway to X.400

```
E:/MHS/IQ
E:/MHS/OQ MHS
```

11.2.3 Contents of gateway.dat for Shared File 1992 Interface

The following is an example of the file gateway.dat for Agents that use the Shared File 1992 interface:

```
/agent_name/iq
/agent_name/oq gateway_name oraddress
```

where:

- *gateway_name* is the name specified when you configured the ISOGATE Access Unit for X.400.
- *oraddress* is the O/R address specified when you configured the ISOGATE Access Unit for X.400.

Edit gateway.dat after it is created by the MTA. You must edit gateway.dat and add the full pathnames and other parameters as specified in the Agent documentation.

Example 11–2 Example gateway.dat for ISOCOR X.400 Router for Lotus Notes

```
t:/notes/iq
t:/notes/oq notes \C=nz\A=nz-ptt\P=acme\O=notes
```

Example 11–2 is an example of the contents of the file gateway.dat for ISOCOR X.400 Router for Lotus Notes.

Example 11–3 is an example of the contents of the file gateway.dat for ISOGATE for cc:Mail.

Example 11–3 Example gateway.dat for ISOGATE for cc:MAIL

```
/path/isogate/iq
/path/isogate/oq ISOGATE \C=nz\A=nz-ptt\P=acme\U=ccmail\O=AAA
```

12
Example User Agent

This chapter describes MTAmail, the example User Agent that is supplied with the MAILbus 400 MTA. You can use this User Agent to experiment with the MTA's routing information, or to generate messages to send to a peer MTA to test a connection.

When submitting a message, MTAmail allows you to request delivery and non-delivery reports, set the priority of the message, and specify multiple recipient names.

When delivering messages, MTAmail displays the recipient and originator details, priority, subject and message contents on the screen. If you choose, MTAmail can also display the structure of the message on the screen. To understand the structure MTAmail displays if you take this option, you need a fuller knowledge of 1992 MHS Standards.

The remaining sections in this chapter explain what you must do to run MTAmail. Section 12.1 lists the prerequisites, Section 12.2 describes how to submit messages, and Section 12.3 describes how to deliver messages and display the structure of a delivered message. Part I of *HP MAILbus 400 MTA Tuning and Problem Solving* explains how the MTA submits and delivers messages.

12.1 Prerequisites

Before you can use MTAmail, you must do the following:

1. Ensure that the MAILbus 400 MTA has been installed and set up.
2. Ensure that the MTA is running, that is, its state is set to ON.
3. Create whatever routing information MTAmail needs for handling your test messages. Note that for sending a test message from an originator to a recipient sharing the same User Agent, you do not need to enter information about peer MTAs or other routing domains; you need only create an Agent entity.
4. Set up some sample O/R addresses in the directory.

Note that MTAmail allows you to specify any number of recipients for a message.

12.2 Submitting a Message to the MTA

After you have installed the MAILbus 400 MTA, MTAmail is in the following location:

Tru64 UNIX

```
/usr/examples/mta/mtamail
```

◆

OpenVMS

```
SYS$COMMON:[SYSTEST]MTA$MTAMAIL.EXE
```

Before you use MTAmail, you also need to define `mtamail` to be a foreign command, for example:

```
mtamail :== $SYS$COMMON:[SYSTEST]MTA$MTAMAIL.EXE
```

◆

When you invoke MTAmail to submit a message, MTAmail prompts you for the information it needs to build the message. When you have entered the necessary information, MTAmail builds the message and submits it to the MTA.

If the message is invalid because it does not have an originator and at least one recipient, MTAmail rejects the message as soon as you submit it. If the message is invalid because it contains incorrect information, for example, invalid O/R addresses, the MTA rejects it and issues a non-delivery report.

When you exit MTAmail, MTAmail submits the message to the MTA. Type Ctrl/D (Tru64 UNIX) or Ctrl/Z (OpenVMS) to exit from MTAmail. If you want to read the message as the recipient, you must invoke MTAmail again, as described in Section 12.3.

Section 12.2.1 gives step by step instructions for submitting a message through MTAmail, and Section 12.2.2 gives a simple example.

12.2.1 Step by Step Instructions for Submitting a Message

You can use either mnemonic addresses or numeric addresses for your MTAmail message. The following instructions assume mnemonic addresses:

1. Invoke MTAmail with the `-s` option as follows:

   `mtamail -s [-r -p`*n*`] `*agent*

   where:

   - *agent* is the name that you used to create an Agent entity at MTA for the MTAmail test Agent. Specify the name after the options.
   - The `-r` option in this command sets report generation to both non-delivery reports and delivery reports. The default is non-delivery reports only.
   - The `-p` *n* option sets the priority of the message; *n* can have the value 1, 2, or 3 — 1=low, 2=normal, 3=high. If the `-p` *n* option is omitted, a priority of 2 (normal) is assumed.

2. Enter the originator address and press Return when the address is complete.
3. Enter the recipient details and press Return when the address is complete. MTAmail prompts for another recipient. You can specify as many recipients as you want. When you have specified addresses for all recipients, press Return at the `Enter Recipient O/R Address` prompt.
4. Specify a Subject.
5. Type the message.
6. Press Return and type Ctrl/D (Tru64 UNIX) or Ctrl/Z (OpenVMS) at the beginning of a new line when your message is complete.

12.2.2 Example

Create and enable an Agent entity called TEST-AGENT of type XAPI, by issuing the following NCL commands at the MTA node:

```
NCL> CREATE MTA AGENT "TEST-AGENT" TYPE=XAPI
NCL> ENABLE MTA AGENT "TEST-AGENT"
```

Create O/R addresses for the originator and the recipient. In this example, the routing domain is `/MTS=ACME`, and the MTA name is `TEST-MTA`.

You need to substitute the name of your routing domain and MTA in the following commands. These are the values you used when setting up your MTA as described in Chapter 8 (Tru64 UNIX) and Chapter 9 (OpenVMS). You can use the O/R addresses given in the example.

```
NCL> CREATE MTS "/MTS=ACME" ORADDRESS "C=NZ"
NCL> CREATE MTS "/MTS=ACME" ORADDRESS "C=NZ; A=NZ-PTT"
NCL> CREATE MTS "/MTS=ACME" ORADDRESS "C=NZ; A=NZ-PTT; P=ACME"
NCL> CREATE MTS "/MTS=ACME" ORADDRESS "C=NZ;A=NZ-PTT;P=ACME;O=ACME"
NCL> CREATE MTS "/MTS=ACME" -
_NCL>           ORADDRESS "C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=WELL"
NCL> CREATE MTS "/MTS=ACME" -
_NCL>           ORADDRESS "C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=WELL;CN=Kim Yip" -
_NCL>           ROUTING INSTRUCTION [ACTION=DELIVER, -
_NCL>           SERVER MTA="TEST-MTA", AGENT="TEST-AGENT"]
NCL> CREATE MTS "/MTS=ACME" -
_NCL>           ORADDRESS "C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=WELL;CN=Mari Best" -
_NCL>           ROUTING INSTRUCTION [ACTION=DELIVER, -
_NCL>           SERVER MTA="TEST-MTA", AGENT="TEST-AGENT"]
```

The rest of this section is a log of the test session (on a Tru64 UNIX system).

```
# /usr/examples/mta/mtamail -s TEST-AGENT
Enter Originator O/R address,
for example, c=nz;a=nz-ptt;p=acme;o=acme;ou1=well;cn=Kim Yip

O/R Address:c=nz;a=nz-ptt;p=acme;o=acme;ou1=well;cn=Kim Yip


Enter Recipient O/R address,
for example, c=nz;a=nz-ptt;p=acme;o=acme;cn=Kim Yip
(Return on 'O/R address' ends)

O/R Address:c=nz;a=nz-ptt;p=acme;o=acme;ou1=well;cn=Mari Best

O/R Address: <Return>
No more recipients

Subject: Test Message

Enter the message, end with <Ctrl>-D at the beginning of a line.

This is a test message sent to show what MTAmail can do.
MTAmail  is the example User Agent supplied with the MAILbus 400
MTA.<Return>
<ctrl/d>

#

Submitting message, please wait ...... Done
```

12.3 Taking Delivery of a Message with MTAmail

Take delivery of a message using MTAmail as follows:

Invoke MTAmail with the `-d` option as follows:

`mtamail -d [-t n] agent`

where:

- *agent* is the name that you used to create an Agent entity at the MTA for the MTAmail test Agent. Specify the name after the options.
- The `-t` *n* option in this command specifies the number of seconds (*n*) that MTAmail should remain connected to the MTA waiting for a message to arrive.

Question to reviewers

What's the default value for n?

MTAmail takes the first message in the Delivery Queue and displays the Originator and Recipient details, priority, subject and message contents on the screen. Example 12–1 shows the message that was submitted in Section 12.2.2.

If there are no messages in the Delivery Queue, and `-t` is not specified, MTAmail displays a message stating that there is no message in the queue.

Example 12–1 Display of a Message Delivered by MTAmail

```
Message Details:

 To:  A=NZ-PTT; C=NZ; CN=TEST-ACCOUNT; O=ACME; OU1=WELL; P=ACME;
 From:  A=NZ-PTT; C=NZ; CN=KIM YIP; O=ACME; OU1=WELL; P=ACME;
 Priority: Normal
 Subject: Test Message

-----  Text Body Part follows -----

This is a test message sent to show what MTAmail can do. MTAmail
is the example User Agent supplied with the MAILbus 400 MTA.
```

12.3.1 Displaying the Structure of a Delivered Message

You can display the entire structure of a delivered message by invoking MTAmail as follows:

```
mtamail -x [-t n] agent
```

In this case, MTAmail takes delivery of the first message in the Delivery Queue, and then displays the message's structure. To interpret this structure, you need a fuller understanding of 1992 MHS Standards.

12.4 Errors Reported by MTAmail

The following sections describe some of the more common errors, and their recovery actions, that might be reported when you use MTAmail.

12.4.1 MH_RC_NO_SUCH_CLIENT

The MTA does not recognize the client name. Make sure that you have created and enabled an Agent entity for MTAmail as described in Section 12.2.2.

12.4.2 OM_NETWORK_ERROR

This error usually occurs because the network link between MTAmail and the MTA has been lost or could not be established. Check the following:

- That MTAmail knows where the MTA is.

  This is described in Section 11.1.

- That the MTA state is ON.
- That the maximum number of Agent connections for the MTA has not been reached. If it has, increase the Maximum Agent Connections attribute in the MTA entity, an entity of the MTA module.

12.4.3 OM_SYSTEM_ERROR

This error can occur for the following reasons:

- When the MTA has a internal problem, for example, with the operating system.

  In this case, the MTA logs an event and returns this error status to MTAmail. Examine the events generated at the MTA to obtain more information about the problem.

- When the MTA has problems connecting to the directory. Check that the DSA state is ON, and that the MTA can access its routing information.
- When the MTS entry does not have a Global Domain Identifiers attribute.

Make sure that you have registered a Global Domain Identifiers attribute for the MTS entity, as described in Section 8.3.2.2 (Tru64 UNIX), or Section 9.4.2.2 (OpenVMS).

12.4.4 OM_WRONG_VALUE_LENGTH

MTAmail has passed an O/R address to the MTA with an attribute that contains a value that exceeds its length constraints. Check the O/R address attribute value length and correct any errors.

12.4.5 OM_WRONG_VALUE_MAKEUP

MTAmail has passed an O/R address to the MTA with an attribute that contains a value with an illegal syntax. Check the O/R address attribute value syntaxes and correct any errors.

A

Postal and Terminal O/R Addresses

This appendix describes the attributes of postal and terminal O/R addresses. These forms of O/R address are used to communicate with postal and terminal routing domains from a MAILbus 400 MTA routing domain. In the case of postal O/R addresses, the MTA routes on a partial O/R address. In the case of the terminal O/R address, the MTA routes on the network address attribute assigned according to telematic standards.

Attributes of the mnemonic and numeric O/R addresses are described in Table 5–1.

This appendix also specifies the attributes required by the MTA to route to these forms of O/R address.

Chapter 7 and the *MTS Module Online Help* describe how to store the attributes that you have planned as part of your naming scheme for these forms of O/R address in the directory.

A.1 Postal O/R Address

A postal O/R address identifies a user by a postal address. This type of O/R address is used to provide interworking between electronic messaging and existing paper mail services. A postal O/R address can be either formatted or unformatted. A formatted postal O/R address is specified by a series of attributes. An unformatted postal O/R address is specified by a single attribute with no defined structure.

Table A–1 lists the attributes of the postal O/R address and indicates whether the attributes are mandatory or conditional. The table also indicates which attributes are used by the MTA to route messages, and the labels that can be used for the attributes.

Table A–1 Postal O/R Address Attributes and Values

Attribute	Label	Used by MTA for Routing	Length and Character Set	Mandatory (M) or Conditional (C)
Attributes Common to all O/R Addresses				
Country Name	C	Yes	2 (alphabetic) 3 (numeric)	M
Administration Domain Name	A	Yes	16	M
Private Domain Name	P	Yes	16	C
Attributes Specific to Postal O/R Addresses				
Physical Delivery Country Name	PD-C	Yes	2 (alphabetic) 3 (numeric)	M
Physical Delivery Service Name	PD-SN	Yes	16 (printable string)	C
Physical Delivery Personal Name	PD-PN	No	30 (printable or teletex string)	C
Extension of Postal O/R Address Components	PD-EA	No	30 (printable or teletex string)	C
Extension of Physical Delivery Address Components	PD-ED	No	30 (printable or teletex string)	C
Physical Delivery Office Number	PD-OFN	No	30 (printable or teletex string)	C
Physical Delivery Office Name	PD-OF	No	30 (printable or teletex string)	C
Physical Delivery Organization Name	PD-O	No	30 (printable or teletex string)	C
Physical Delivery Street Address	PD-S	No	30 (printable or teletex string)	C
Unformatted Postal Address	PD-A1 to PD-A6	No	30 each (printable or teletex string)	M[1]

[1]Mandatory for unformatted postal O/R address form. For the formatted postal O/R address, the other postal O/R address attributes are used.

(continued on next page)

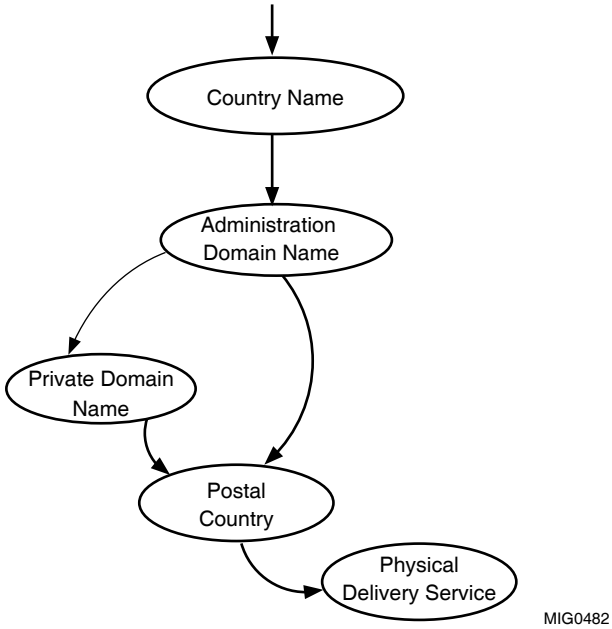
Table A–1 (Cont.) Postal O/R Address Attributes and Values

Attribute	Label	Used by MTA for Routing	Length and Character Set	Mandatory (M) or Conditional (C)
Attributes Specific to Postal O/R Addresses				
Local Postal Attributes	PD-L	No	30 (printable or teletex string)	C
Unique Postal Name	PD-U	No	30 (printable or teletex string)	C
Poste Restante Address	PD-R	No	30 (printable or teletex string)	C
Post Office Box Address	PD-B	No	30 (printable or teletex string)	C
Postal Code	PD-PC	No	16 (printable or numeric string)	M

Note that when planning postal O/R addresses, you include only those attributes that the MTA uses for routing. Define partial postal O/R addresses that describe the community of users served by each Physical Delivery Access Unit (PDAU) in your network. You then add a routing instruction that indicates how an MTA routes to the PDAU. The remaining attributes are ignored by MAILbus 400 MTAs when present in an O/R address. The only exception to this is where a redirection address is specified which needs to include these attributes.

Figure A–1 shows the hierarchy of the postal form of O/R address as it is represented in the directory.

Figure A–1 How Postal O/R Addresses are Constructed



A.2 Terminal O/R Address

A terminal O/R address identifies a user by a network address and, where appropriate, a terminal type. This type of O/R address is used to provide interworking between electronic messaging and existing telematic services, for example, telex and teletex. A terminal O/R address can also identify the ADMD through which the terminal is accessed, using the Administration Domain Name and Country Name attributes.

If the terminal is a Telematic terminal, the terminal O/R address comprises the terminal's network address and possibly, using the Terminal Type and Terminal Identifier attributes, its terminal type and identifier. If the terminal is a Telex terminal, the terminal O/R address comprises the terminal's Telex number.

In the case of a telex terminal, the Network Address is expressed as an X.121 address, and the Terminal Identifier attribute provides the Telex answer-back string for the subscriber.

Table A–2 lists the attributes of a terminal O/R address and indicates whether the attributes are mandatory or conditional. The table also indicates which attributes are used by the MTA to route messages and the labels that can be used for the attributes.

Table A–2 Terminal O/R Address Attributes and Values

Attribute	Label	Used by MTA for Routing	Length and Character Set	Mandatory (M) or Conditional (C)
Attributes Common to all O/R Addresses				
Country Name	C	Yes	2 (alphabetic) 3 (numeric)	C
Administration Domain Name	A	Yes	16	C
Private Domain Name	P	Yes	16	C

(continued on next page)

Table A–2 (Cont.) Terminal O/R Address Attributes and Values

Attribute	Label	Used by MTA for Routing	Length and Character Set	Mandatory (M) or Conditional (C)
Attributes Common to all O/R Addresses				
Domain Defined 1 Domain Defined 2 Domain Defined 3 Domain Defined 4	DDA:	No	8 (type): 128 (value) (teletex or printable)[1]	C
Attributes Specific to Terminal O/R Addresses				
Terminal Type	T-TY	Yes	TLX (Telex) TTX (Teletex) G3FAX (G3-facsimile) G4FAX (G4-facsimile) IA5 (IA5-terminal) VTX (Videotex)	C
Network Address	X.121	Yes	15 (numeric string)	M
Terminal Identifier	T-ID	No	24 (printable string)	C
Extended Network Address	E.164 or PSAP	No	Unlimited (numeric) for E.164 or Presentation address[2] for PSAP	C
Attributes Intended to Provide Descriptive Information for Terminal O/R addresses.				
Organization Name	O	No	64 (teletex or printable)	C
Organizational Unit 1	OU1	No	32 (teletex or printable)	C
Organizational Unit 2	OU2	No	32 (teletex or printable)	C
Organizational Unit 3	OU3	No	32 (teletex or printable)	C
Organizational Unit 4	OU4	No	32 (teletex or printable)	C
Common Name	CN	No	64 (teletex or printable)	C

[1]To enter a DDA Type in the directory that contains an equals sign "=", you need to specify an escape sequence. For the MAILbus 400 MTA this is done by doubling the equals sign; for example, DDA type of fax=fax22 is entered as: DDA:fax= =fax22=+44 000 000000

[2]Refer to the DECnet-Plus documentation set for details of the format of a Presentation address.

(continued on next page)

Table A–2 (Cont.) Terminal O/R Address Attributes and Values

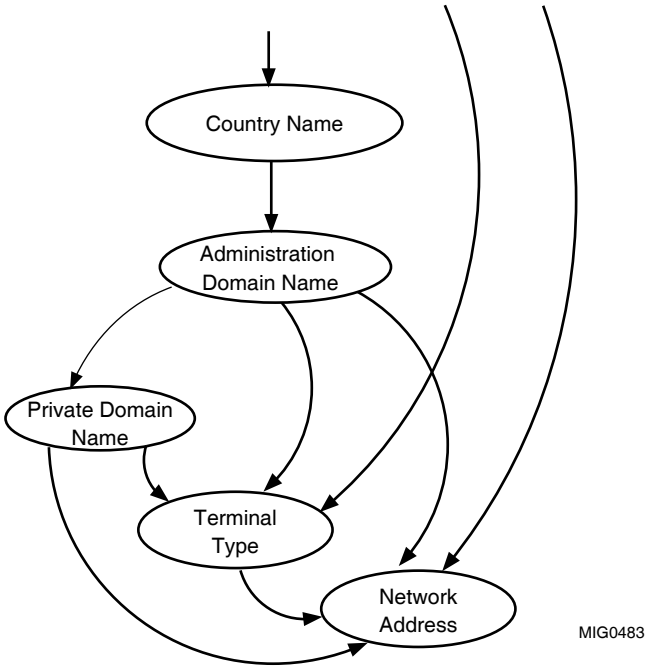
Attribute	Label	Used by MTA for Routing	Length and Character Set	Mandatory (M) or Conditional (C)
Attributes Intended to Provide Descriptive Information for Terminal O/R addresses.				
Personal Name comprising:	-	No	-	C
Given Name	G	-	16 (teletex or printable)	C
Initials	I	-	5 (teletex or printable)	C
Surname	S	-	40 (teletex or printable)	M[3]
Generation Qualifier	Q	-	3 (teletex or printable)	C
Unformatted Postal Address	PD-A1 to PD-A6	No	30 each (printable or teletex string)	C

[3]When a Personal Name is used this attribute is mandatory.

Note that when planning terminal O/R addresses, you only include those attributes that the MTA uses for routing. Define complete or partial terminal O/R addresses that describe the community of users served by the terminal type, or network address. The remaining attributes are ignored when present in an O/R address. The only exception to this is where a redirection address is specified which includes these attributes.

Figure A–2 shows the hierarchy of the terminal form of O/R address as it is represented in the directory.

Figure A–2 How Terminal O/R Addresses are Constructed



B

MTS Script for ACME

This appendix contains the MTS script for the ACME Shoe Corporation, used as an example throughout this guide.

This script is also available online. The location of the script is given in the list of files in *HP MAILbus 400 MTA Tuning and Problem Solving*.

```
*5I
 #%COPYRIGHT_START%
 #
*7D
 #    © Copyright 2006 Hewlett-Packard Development Company, L.P.
*7E
*7I
*15D
 #    © Copyright 2006 Hewlett-Packard Development Company, L.P.
*15E
*15I
*17D
 #    © Copyright 2006 Hewlett-Packard Development Company, L.P.
*17E
*17I
 #    © Copyright 2006 Hewlett-Packard Development Company, L.P.
*17E
*15E
*7E
 #
 #    Restricted Rights: Use, duplication, or disclosure by the U.S.
 #    Government is subject to restrictions as set forth in subparagraph (C)
 #    (1) (ii) of DFARS 252.227-7013, or in FAR 52.227-19, or in FAR
 #    52.227-14 Alt. III, as applicable. Unpublished rights reserved under
 #    applicable copyright laws.
 #
 #    This software is proprietary to and embodies the confidential
 #    technology of Hewlett-Packard. Possession, use, or
 #    copying of this software and media is authorized only pursuant to a
 #    valid written license from HP or an authorized sublicensor.
 #
 #%COPYRIGHT_END%
 !
```

```
*5E
 ! This file contains commands to create the routing information
 ! for the routing domain ACME, used as an example in the MAILbus 400
 ! MTA Documentation.
 !
 ! Create the ACME routing domain entry in the directory
 ! and assign ACME's GDIs.
 !
*2D
 CREATE MTS "/C=NZ/O=ACME/MTS=ACME"
*2E
*2I
*4D
 CREATE MTS "/C=NZ/O=ACME/MTS=ACME" PASSWORD "unforgettable"
*4E
*4I
*15D
 CREATE MTS "/C=NZ/O=ACME/MTS=ACME" PASSWORD -
*15E
*15I
 CREATE MTS "/MTS=ACME" PASSWORD -
*15E
 "unforgettable-mts-password"
*4E
*2E
*15D
 SET MTS "/C=NZ/O=ACME/MTS=ACME" DESCRIPTION "The ACME Routing Domain"
 SET MTS "/C=NZ/O=ACME/MTS=ACME" GLOBAL DOMAIN IDENTIFIERS -
*15E
*15I
 SET MTS "/MTS=ACME" DESCRIPTION "The ACME Routing Domain"
 SET MTS "/MTS=ACME" GLOBAL DOMAIN IDENTIFIERS -
*15E
  {"C=NZ;A=NZ-PTT;P=ACME","C=NZ;A=AU-PTT;P=ACME"}
*2D
*2E
*2I
```

```
 ! You might also need authority to create the MTS entity
*15D
 ! CREATE MTS "/C=NZ/O=ACME/MTS=ACME" PASSWORD "unforgettable" -
*15E
*15I
 ! CREATE MTS "/MTS=ACME" PASSWORD "unforgettable" -
*15E
 ! AUTHORIZING USER "/C=NZ/O=ACME/CN=Directory Manager"
*6D
 ! AUTHORIZING PASSWORD "chocolate"
*6E
*6I
 ! AUTHORIZING PASSWORD "chocolate"
*6E
*2E
 !
 ! Create the Auckland area in the ACME routing domain.
 !

*15D
 CREATE MTS "/C=NZ/O=ACME/MTS=ACME" AREA "AUCK"
 SET MTS "/C=NZ/O=ACME/MTS=ACME" AREA "AUCK" -
*15E
*15I
 CREATE MTS "/MTS=ACME" AREA "AUCK"
 SET MTS "/MTS=ACME" AREA "AUCK" -
*15E
  DESCRIPTION "The Auckland Area within ACME"

 !
 ! Assign an area server to the AUCK area so ACME can use area
 !  routing.
 !

*15D
 SET MTS "/C=NZ/O=ACME/MTS=ACME" AREA "AUCK" SERVER MTA "AUCK.MTA-SET1"
*15E
*15I
 SET MTS "/MTS=ACME" AREA "AUCK" SERVER MTA "AUCK.MTA-SET1"
*15E

 !
 ! Create the Wellington area in the ACME routing domain.
 !

*15D
 CREATE MTS "/C=NZ/O=ACME/MTS=ACME" AREA "WELL"
 SET MTS "/C=NZ/O=ACME/MTS=ACME" AREA "WELL" -
*15E
*15I
 CREATE MTS "/MTS=ACME" AREA "WELL"
 SET MTS "/MTS=ACME" AREA "WELL" -
*15E
  DESCRIPTION "The Wellington Area within ACME"
```

```
 !
 ! Assign an area server to the WELL area so ACME can use area
 !  routing.
*15D
 SET MTS "/C=NZ/O=ACME/MTS=ACME" AREA "WELL" SERVER MTA "WELL.MTA-NODE7"
*15E
*15I
 SET MTS "/MTS=ACME" AREA "WELL" SERVER MTA "WELL.MTA-NODE7"
*15E

 !
 ! MTA entries are created when an individual MTA is installed and set
 ! up.
 !

 !
 ! Create an MTA set as the area server for AUCK.
 !

*15D
 CREATE MTS "/C=NZ/O=ACME/MTS=ACME" MTA "AUCK.MTA-SET1" TYPE MTA SET
 SET MTS "/C=NZ/O=ACME/MTS=ACME" MTA "AUCK.MTA-SET1" -
*15E
*15I
 CREATE MTS "/MTS=ACME" MTA "AUCK.MTA-SET1" TYPE MTA SET
 SET MTS "/MTS=ACME" MTA "AUCK.MTA-SET1" -
*15E
  MEMBERS {"AUCK.MTA-NODE11","AUCK.MTA-NODE12"}

 !
 ! Create the MACE domain entry.
 !

*15D
 CREATE MTS "/C=NZ/O=ACME/MTS=ACME" DOMAIN "MACE" -
*15E
*15I
 CREATE MTS "/MTS=ACME" DOMAIN "MACE" -
*15E
*2I
*4D
  PASSWORD "mydomain" -
*4E
*2E
  DIFFERENT CCITT DOMAIN TRUE
*15D
 SET MTS "/C=NZ/O=ACME/MTS=ACME" DOMAIN "MACE" DESCRIPTION -
*15E
*15I
 SET MTS "/MTS=ACME" DOMAIN "MACE" DESCRIPTION -
*15E
  "The MACE routing domain; direct connection"
```

```
 !
 ! Assign a routing instruction to show how the MACE routing
 ! domain is served.
 !
*15D
 SET MTS "/C=NZ/O=ACME/MTS=ACME" DOMAIN "MACE" ROUTING INSTRUCTION -
*15E
*15I
 SET MTS "/MTS=ACME" DOMAIN "MACE" ROUTING INSTRUCTION -
*15E
  [ACTION=TRANSFER TO DOMAIN, BOUNDARY MTA="WELL.MTA-NODE6"]
 !
 ! Create the EMAC domain entry.
 !
*15D
 CREATE MTS "/C=NZ/O=ACME/MTS=ACME" DOMAIN "EMAC" -
*15E
*15I
 CREATE MTS "/MTS=ACME" DOMAIN "EMAC" -
*15E
*2I
*4D
  PASSWORD "yourdomain" -
*4E
*2E
  DIFFERENT CCITT DOMAIN TRUE
*15D
 SET MTS "/C=NZ/O=ACME/MTS=ACME" DOMAIN "EMAC" DESCRIPTION -
*15E
*15I
 SET MTS "/MTS=ACME" DOMAIN "EMAC" DESCRIPTION -
*15E
  "The EMAC routing domain; indirect connection through MACE"
 !
 ! Assign a routing instruction to show how the EMAC routing
*15D
 ! domain is served.
*15E
*15I
*18D
 .! domain is served.
*18E
*18I
 ! domain is served.
*18E
*15E
 !
```

```
*15D
 SET MTS "/C=NZ/O=ACME/MTS=ACME" DOMAIN "EMAC" ROUTING INSTRUCTION -
*15E
*15I
 SET MTS "/MTS=ACME" DOMAIN "EMAC" ROUTING INSTRUCTION -
*15E
  [ACTION=TRANSFER THROUGH DOMAIN, SERVER DOMAIN="MACE"]

 !
 ! Create an SMTP domain entry named D1.
 !

*2D
 CREATE MTS "/C=NZ/O=ACME/MTS=ACME" DOMAIN "D1"
*2E
*2I
*15D
 CREATE MTS "/C=NZ/O=ACME/MTS=ACME" DOMAIN "D1" -
*15E
*15I
 CREATE MTS "/MTS=ACME" DOMAIN "D1" -
*15E
*4D
  PASSWORD "anotherdomain"
*4E
*4I
  PASSWORD "an-smtp-domain"
*4E
*2E

 !
 ! Assign a routing instruction to show how the SMTP routing
 ! domain is served.
 !

*15D
 SET MTS "/C=NZ/O=ACME/MTS=ACME" DOMAIN "D1" ROUTING INSTRUCTION -
*15E
*15I
*18D
 .SET MTS "/MTS=ACME" DOMAIN "D1" ROUTING INSTRUCTION -
*18E
*18I
 SET MTS "/MTS=ACME" DOMAIN "D1" ROUTING INSTRUCTION -
*18E
*15E
  [ACTION=DELIVER, DELIVERING MTA="WELL.MTA-NODE8", -
  AGENT="SMTP-GW"]

 !
 ! Create an SMTP Foreign Address entry.
 !
```

```
*15D
 CREATE MTS "/C=NZ/O=ACME/MTS=ACME" DOMAIN "D1" -
*15E
*15I
 CREATE MTS "/MTS=ACME" DOMAIN "D1" -
*15E
  FOREIGN ADDRESS "D1.NZ"
*15D
 SET MTS "/C=NZ/O=ACME/MTS=ACME" DOMAIN "D1" -
*15E
*15I
 SET MTS "/MTS=ACME" DOMAIN "D1" -
*15E
  FOREIGN ADDRESS "D1.NZ" -
  DESCRIPTION "Partial O/R address representing the SMTP routing domain", -
  TRANSLATION "C=NZ; A=NZ-PTT; P=D1"

 !
 ! Create ACME GDI O/R Address attributes.
 !

*15D
 CREATE MTS "/C=NZ/O=ACME/MTS=ACME" ORADDRESS  "C=NZ"
 CREATE MTS "/C=NZ/O=ACME/MTS=ACME" ORADDRESS  "C=NZ; A=NZ-PTT"
 CREATE MTS "/C=NZ/O=ACME/MTS=ACME" ORADDRESS  "C=NZ; A=NZ-PTT; P=ACME"
*15E
*15I
 CREATE MTS "/MTS=ACME" ORADDRESS  "C=NZ"
 CREATE MTS "/MTS=ACME" ORADDRESS  "C=NZ; A=NZ-PTT"
 CREATE MTS "/MTS=ACME" ORADDRESS  "C=NZ; A=NZ-PTT; P=ACME"
*15E
 !
*15D
 CREATE MTS "/C=NZ/O=ACME/MTS=ACME" ORADDRESS  "C=AU"
 CREATE MTS "/C=NZ/O=ACME/MTS=ACME" ORADDRESS  "C=AU; A=AU-PTT"
 CREATE MTS "/C=NZ/O=ACME/MTS=ACME" ORADDRESS  "C=AU; A=AU-PTT; P=ACME"
*15E
*15I
 CREATE MTS "/MTS=ACME" ORADDRESS  "C=AU"
 CREATE MTS "/MTS=ACME" ORADDRESS  "C=AU; A=AU-PTT"
 CREATE MTS "/MTS=ACME" ORADDRESS  "C=AU; A=AU-PTT; P=ACME"
*15E

 !
 ! Set alias for GDI O/R Address attributes.
 !
```

```
*15D
 SET MTS "/C=NZ/O=ACME/MTS=ACME" ORADDRESS  "C=AU; A=AU-PTT; P=ACME" -
*15E
*15I
 SET MTS "/MTS=ACME" ORADDRESS  "C=AU; A=AU-PTT; P=ACME" -
*15E
  TYPE ALIAS, ALIAS TARGET "C=NZ; A=NZ-PTT; P=ACME"

 !
 ! Create alias for country name.
 !

*15D
 CREATE MTS "/C=NZ/O=ACME/MTS=ACME" ORADDRESS  "C=530" -
*15E
*15I
 CREATE MTS "/MTS=ACME" ORADDRESS  "C=530" -
*15E
  TYPE ALIAS, ALIAS TARGET "C=NZ"

 !
 ! Create remaining ACME O/R address hierarchy.
 !

*15D
 CREATE MTS "/C=NZ/O=ACME/MTS=ACME" -
*15E
*15I
 CREATE MTS "/MTS=ACME" -
*15E
  ORADDRESS "C=NZ;A=NZ-PTT;P=ACME;O=ACME"
*15D
 CREATE MTS "/C=NZ/O=ACME/MTS=ACME" -
*15E
*15I
 CREATE MTS "/MTS=ACME" -
*15E
  ORADDRESS "C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=AUCK"
*15D
 CREATE MTS "/C=NZ/O=ACME/MTS=ACME" -
*15E
*15I
 CREATE MTS "/MTS=ACME" -
*15E
  ORADDRESS "C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=WELL"

 !
 ! Create ACME individual O/R addresses.
 !
```

```
*15D
 CREATE MTS "/C=NZ/O=ACME/MTS=ACME" ORADDRESS -
*15E
*15I
 CREATE MTS "/MTS=ACME" ORADDRESS -
*15E
  "C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=WELL;CN=Clare Roberts"
*15D
 CREATE MTS "/C=NZ/O=ACME/MTS=ACME" ORADDRESS -
*15E
*15I
 CREATE MTS "/MTS=ACME" ORADDRESS -
*15E
  "C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=WELL;CN=Jane Underwood-Murray"
*15D
 CREATE MTS "/C=NZ/O=ACME/MTS=ACME" ORADDRESS -
*15E
*15I
 CREATE MTS "/MTS=ACME" ORADDRESS -
*15E
  "C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=AUCK;CN=William Laurence"
*15D
 CREATE MTS "/C=NZ/O=ACME/MTS=ACME" ORADDRESS -
*15E
*15I
 CREATE MTS "/MTS=ACME" ORADDRESS -
*15E
  "C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=AUCK;CN=Fred Smith"

 !
 ! Commands to create alternative O/R addresses for William Laurence.
 !
```

```
*15D
 CREATE MTS "/C=NZ/O=ACME/MTS=ACME" ORADDRESS -
*15E
*15I
 CREATE MTS "/MTS=ACME" ORADDRESS -
*15E
  "C=NZ;A=NZ-PTT; P=ACME;O=ACME;OU1=AUCK;CN=BILL LAURENCE" -
  TYPE ALIAS, ALIAS TARGET -
  "C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=AUCK;CN=WILLIAM LAURENCE"
*15D
 CREATE MTS "/C=NZ/O=ACME/MTS=ACME" ORADDRESS -
*15E
*15I
 CREATE MTS "/MTS=ACME" ORADDRESS -
*15E
  "C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=AUCK;CN=WILLIAM LAWRENCE" -
  TYPE ALIAS, ALIAS TARGET -
  "C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=AUCK;CN=WILLIAM LAURENCE"
*15D
 CREATE MTS "/C=NZ/O=ACME/MTS=ACME" ORADDRESS  -
*15E
*15I
 CREATE MTS "/MTS=ACME" ORADDRESS  -
*15E
  "C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=AUCK;CN=BILL LAWRENCE" -
  TYPE ALIAS, ALIAS TARGET -
  "C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=AUCK;CN=WILLIAM LAURENCE"

 !
 ! Personal Name for Jane Underwood-Murray.
 !

*15D
 SET MTS "/C=NZ/O=ACME/MTS=ACME" ORADDRESS -
*15E
*15I
 SET MTS "/MTS=ACME" ORADDRESS -
*15E
  "C=NZ; A=NZ-PTT; P=ACME; O=ACME; OU1=WELL; CN=Jane Underwood-Murray" -
  PERSONAL NAME "S=UNDERWOOD;G=JANE"

 !
 ! Routing Information for individual O/R address.
 !
 !
 ! Clare Roberts served by the registered Agent, UA1.
 !
```

```
*15D
 SET MTS "/C=NZ/O=ACME/MTS=ACME" ORADDRESS -
*15E
*15I
 SET MTS "/MTS=ACME" ORADDRESS -
*15E
  "C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=WELL;CN=Clare Roberts" -
  ROUTING INSTRUCTION [ACTION=DELIVER, -
  SERVER MTA="WELL.MTA-NODE7", AGENT="UA1", -
  DEFINITIVE ORADDRESS = -
  "C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=WELL;CN=Clare Roberts"]
 !
 ! Jane Underwood-Murray served by the registered Agent, UA1.
 ! In this case the routing instruction is commented out as a
 ! redirection is set.
 !
*15D
 !SET MTS "/C=NZ/O=ACME/MTS=ACME" ORADDRESS -
*15E
*15I
 !SET MTS "/MTS=ACME" ORADDRESS -
*15E
 ! "C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=WELL;CN=Jane Underwood-Murray" -
 ! ROUTING INSTRUCTION [ACTION=DELIVER, -
 ! SERVER MTA="WELL.MTA-NODE7", AGENT="UA1", -
 ! DEFINITIVE ORADDRESS= -
 ! "C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=WELL;CN=Jane Underwood-Murray"]
 !
 !
 ! Temporarily redirect mail for Jane Underwood-Murray to
 ! WELL Secretary
 !

*15D
 SET MTS "/C=NZ/O=ACME/MTS=ACME" ORADDRESS -
*15E
*15I
 SET MTS "/MTS=ACME" ORADDRESS -
*15E
  "C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=WELL;CN=Jane Underwood-Murray" -
  ROUTING INSTRUCTION [ACTION=RECIPIENT REDIRECT, -
  REDIRECTION="C=NZ;A=NZ-PTT;P=ACME;OU1=WELL;CN=WELL Secretary"]
 !
 ! William Laurence served by the registered Agent, UA2.
 !
```

```
*15D
 SET MTS "/C=NZ/O=ACME/MTS=ACME" ORADDRESS -
*15E
*15I
 SET MTS "/MTS=ACME" ORADDRESS -
*15E
  "C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=AUCK;CN=William Laurence" -
  ROUTING INSTRUCTION [ACTION=DELIVER, -
  SERVER MTA="AUCK.MTA-NODE1", AGENT="UA2", -
  DEFINITIVE ORADDRESS = -
  "C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=AUCK;CN=WILLIAM LAURENCE"]
 !
 ! Fred Smith served by an unregistered Agent.
 !

*15D
 SET MTS "/C=NZ/O=ACME/MTS=ACME" ORADDRESS -
*15E
*15I
 SET MTS "/MTS=ACME" ORADDRESS -
*15E
  "C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=AUCK;CN=Fred Smith" -
  ROUTING INSTRUCTION [ACTION=DELIVER, -
  SERVER MTA="AUCK.MTA-NODE1", AGENT="", -
  DEFINITIVE ORADDRESS= -
  "C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=AUCK;CN=Fred Smith"], -
  PASSWORD = "CHEESE"
 !
 ! O/R address and Routing Information for the community of users in
 ! the D1 routing domain.
 !

*15D
 CREATE MTS "/C=NZ/O=ACME/MTS=ACME" ORADDRESS -
*15E
*15I
 CREATE MTS "/MTS=ACME" ORADDRESS -
*15E
  "C=NZ;A=NZ-PTT;P=D1" -
  ROUTING INSTRUCTION [ACTION=TRANSFER TO DOMAIN, -
  SERVER DOMAIN="D1"]

 !
 ! The foreign address translation for Claudia Hofmann.
 !
```

```
*15D
 CREATE MTS "/C=NZ/O=ACME/MTS=ACME" ORADDRESS -
*15E
*15I
 CREATE MTS "/MTS=ACME" ORADDRESS -
*15E
  "C=NZ;A=NZ-PTT;P=D1;CN=Claudia Hofmann" -
  TRANSLATIONS {[DOMAIN="D1", FOREIGN ADDRESS = -
  "ch@d1.node-grape"]}

 !
 ! Content Information for MAILbus 400 SMTP Gateway
 !
*15D
 SET MTS "/C=NZ/O=ACME/MTS=ACME" ORADDRESS -
*15E
*15I
 SET MTS "/MTS=ACME" ORADDRESS -
*15E
  "C=NZ;A=NZ-PTT;P=D1" -
*19D
  CONTENT INFORMATION -
  [MAXIMUM CONTENT LENGTH=1000, -
  CONTENT TYPES={"{1 3 12 2 1011 5 5 0 1 2}"}, -
  ENCODED INFORMATION TYPES={ -
  "{2 6 3 4 2}", -
  "{2 6 3 4 5}", -
  "{1 3 12 1011 1 3 1}", -
  "{1 3 12 2 1011 5 5 2 0 14}"}]
```

```
*19E
*19I
   CONTENT INFORMATION [MAXIMUM CONTENT LENGTH = 0, -
   CONTENT TYPES = ( -
   "{1 3 12 2 1011 5 5 0 1 22}"),-
   ENCODED INFORMATION TYPES = ( -
   "{1 0 10021 7 1 0 6}", -
   "{1 0 10021 7 1 0 100}", -
   "{1 0 10021 7 1 0 1}", -
   "{1 0 10021 7 1 0 77}", -
   "{1 0 10021 7 1 0 101}", -
   "{1 0 10021 7 1 0 109}", -
   "{1 0 10021 7 1 0 110}", -
   "{1 0 10021 7 1 0 144}", -
   "{1 0 10021 7 1 0 127}", -
   "{1 0 10021 7 1 0 126}", -
   "{1 0 10021 7 1 0 138}", -
   "{1 0 10021 7 1 0 148}", -
   "{1 0 10021 7 1 0 2}", -
   "{2 6 3 4 2}", -
   "{1 3 12 1011 1 3 1}", -
   "{1 3 12 2 1011 5 5 1 0}")]
*19E
 !
 ! Content information for Clare Roberts' O/R address.
 !
*15D
 SET MTS "/C=NZ/O=ACME/MTS=ACME" ORADDRESS -
*15E
*15I
 SET MTS "/MTS=ACME" ORADDRESS -
*15E
  "C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=WELL;CN=Clare Roberts" -
  CONTENT INFORMATION -
  [MAXIMUM CONTENT LENGTH=1000, -
  CONTENT TYPES={"{1 3 12 2 1011 5 5 0 1 22}", -
  "{1 3 12 2 1011 5 5 0 0}"}, -
  ENCODED INFORMATION TYPES={ -
  "{1 0 10021 7 1 0 102}", -
  "{1 0 10021 7 1 0 103}", -
  "{1 0 10021 7 1 0 106}", -
  "{1 0 10021 7 1 0 107}"}]
 !
 ! Create a distribution list.
 !
```

```
*15D
 CREATE MTS "/C=NZ/O=ACME/MTS=ACME" ORADDRESS -
*15E
*15I
 CREATE MTS "/MTS=ACME" ORADDRESS -
*15E
  "C=NZ;A=NZ-PTT;P=ACME;O=ACME;CN=DIST-LIST1" -
  TYPE=DISTRIBUTION LIST, MEMBERS { -
  "C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=WELL;CN=Clare Roberts", -
  "C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=AUCK;CN=William Laurence", -
  "C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=WELL;CN=Jane Underwood", -
  "C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=AUCK;CN=Fred Smith" }

*15D
 ADD MTS "/C=NZ/O=ACME/MTS=ACME" ORADDRESS -
*15E
*15I
 ADD MTS "/MTS=ACME" ORADDRESS -
*15E
  "C=NZ; A=NZ-PTT; P=ACME; O=ACME; CN=DIST-LIST1"  -
  MEMBERS { -
  "C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=WELL;CN=Henry Petra", -
  "C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=WELL;CN=Simon Prior", -
  "C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=WELL;CN=Freda Turner", -
  "C=NZ; A=NZ-PTT;P=ACME;O=ACME;OU1=WELL;CN=Cathy Barnes" }
```

C

MTS Module Entities and Attributes

This appendix lists the characteristic attributes of each of the following entities of the MTS module:

- MTS entity (Table C–1)
- MTA entity (Table C–2)
- Area entity (Table C–3)
- ORaddress entity (Table C–4)
- Domain entity (Table C–5)
- Foreign Address entity (Table C–6)

Unless otherwise specified all attributes can be set and displayed using the Set and Show directives. Each table also gives the attribute's datatype or value. See the *MTS Module Online Help* for a detailed description of all attributes and their syntaxes.

In the tables, **THIS TYPEFACE** indicates specific attributes and values that you type in as part of the Set or Show directive. The normal typeface describes the attribute, value or datatype.

Table C–1 MTS Entity Attributes

Attribute	Datatype or Value
Identifier	X.500 Distinguished Name
PASSWORD[1]	Printable string
GLOBAL DOMAIN IDENTIFIERS	Set of partial O/R addresses comprising Country Name, Administration Domain Name and, optionally, Private Domain Name

[1]This attribute cannot be displayed.

(continued on next page)

Table C–1 (Cont.) MTS Entity Attributes

Attribute	Datatype or Value
DESCRIPTION	Teletex string

Table C–2 MTA Entity Attributes

Attribute	Datatype or Value
Identifier	Printable string
TYPE [1]	SINGLE MTA or MTA SET
PRESENTATION ADDRESS	See appropriate DECnet-Plus NCL reference documentation for syntax. Value determined automatically as part of MTA setup.
SESSION ADDRESS	See appropriate DECnet-Plus NCL reference documentation for syntax. Value determined automatically as part of MTA setup.
PASSWORD[2]	Printable string
MEMBERS[3]	Set of MTA identifiers
DESCRIPTION	Teletex string

[1]Where TYPE not specified, the MTA assumes a value of SINGLE MTA.

[2]This attribute cannot be displayed.

[3]You can also use the Add and Remove commands for this attribute.

Table C–3 Area Entity Attributes

Attribute	Datatype or Value
Identifier	Printable string
SERVER MTA	MTA identifier
DESCRIPTION	Teletex string

Table C–4 ORaddress Entity Attributes

Attribute	Datatype or Value
Identifier	O/R address

(continued on next page)

Table C–4 (Cont.) ORaddress Entity Attributes

Attribute	Datatype or Value	
TYPE [1]	USER, DISTRIBUTION LIST OR ALIAS	
ROUTING INSTRUCTION	ACTION taken by an MTA when routing to this O/R address.	
	One of[2]:	
	ACTION	DELIVER
	SERVER MTA	MTA identifier
	AGENT	Agent name
	DEFINITIVE ORADDRESS	O/R address
	ACTION	TRANSFER TO DOMAIN
	SERVER DOMAIN	Domain identifier
	ACTION	RECIPIENT REDIRECT
	REDIRECTION	O/R address
	ACTION	MD REDIRECT
	REDIRECTION	O/R address
	ACTION	NONDELIVER

[1]Where TYPE not specified, the MTA assumes a value of USER.

[2]Also applies to MNEMONIC ROUTING INSTRUCTION, NUMERIC ROUTING INSTRUCTION, TERMINAL ROUTING INSTRUCTION and POSTAL ROUTING INSTRUCTION.

(continued on next page)

Table C–4 (Cont.) ORaddress Entity Attributes

Attribute	Datatype or Value	
CONTENT INFORMATION	Details of Content Types, Encoded Information Types and Maximum Content Lengths applicable to this O/R address.	
	Set of[3]:	
	MAXIMUM CONTENT LENGTH	Integer
	CONTENT TYPES	Set of object identifiers, see Part II of *HP MAILbus 400 MTA Tuning and Problem Solving* for list of object identifier values.
	ENCODED INFORMATION TYPES	Set of object identifiers, see Part II of *HP MAILbus 400 MTA Tuning and Problem Solving* for list of object identifier values.
ALIAS TARGET[4]	O/R address	
MAY CROSS CCITT BOUNDARIES[5]	TRUE or FALSE	
MEMBERS[6,7]	Set of O/R addresses	
PASSWORD[8]	Printable String	
PERSONAL NAME	O/R address	
TRANSLATIONS[7]	List of other mail address formats for individual with this O/R address.	
	Set of:	
	DOMAIN	Domain identifier
	FOREIGN ADDRESS	Printable or teletex string
DESCRIPTION	Teletex string	

[3]Also applies to MNEMONIC CONTENT INFORMATION, NUMERIC CONTENT INFORMATION, TERMINAL CONTENT INFORMATION and POSTAL CONTENT INFORMATION.

[4]Attribute used when Type=Alias.

[5]Where attribute not specified, the MTA assumes a value of TRUE.

[6]Attribute used when Type=Distribution List.

[7]You can also use the Add and Remove commands for this attribute.

[8]This attribute cannot be displayed.

Table C–5 Domain Entity Attributes

Attribute	Datatype or Value	
Identifier	Printable string	
PASSWORD[1]	Printable string	
ROUTING INSTRUCTION	ACTION to be taken by an MTA when routing to this routing domain.	
	One of:	
	ACTION	DELIVER
	DELIVERING MTA	MTA identifier
	AGENT	Agent name
	ACTION	TRANSFER TO DOMAIN
	BOUNDARY MTA	MTA identifier
	ACTION	TRANSFER THROUGH DOMAIN
	SERVER DOMAIN	Domain identifier
DIFFERENT CCITT DOMAIN[2]	TRUE OR FALSE	
DESCRIPTION	Teletex string	

[1]This attribute cannot be displayed.

[2]Where attribute not specified, the MTA assumes a value of FALSE.

Table C–6 Foreign Address Entity Attributes

Attribute	Datatype or Value
Identifier	Printable or teletex string
TRANSLATION	O/R address
DESCRIPTION	Teletex string

D

Country Codes

The ISO International Standard 3166 defines the alphabetic codes for the representation of country names. The CCITT recommendation X.121 defines the numeric codes for the representation of country names. You can specify a country either by its two-character alphabetic code or by its three-character numeric code. Both sets of codes are listed in Table D–1.

There is an NCL script installed as part of the MTA that contains the commands to create the two-character alphabetic country codes for these countries in the directory. Instructions that describe how to use the script in your routing domain are provided in the script. Refer to *HP MAILbus 400 MTA Tuning and Problem Solving* for the location of this NCL script.

Table D–1 Country Codes

COUNTRY	CODES	
	Alphabetic	**Numeric**
Afghanistan	AF	412
Albania	AL	276
Algeria	DZ	603
American Samoa	AS	544
Andorra	AD	
Angola	AO	631
Anguilla	AI	
Antarctica	AQ	
Antigua and Barbuda	AG	344
Argentina	AR	722
Armenia	AM	
Aruba	AW	
Australia	AU	505
Austria	AT	232

(continued on next page)

Table D–1 (Cont.) Country Codes

COUNTRY	CODES	
	Alphabetic	**Numeric**
Azerbaijan	AZ	400
Bahamas	BS	364
Bahrain	BH	426
Bangladesh	BD	470
Barbados	BB	342
Belarus	BY	
Belgium	BE	206
Belize	BZ	702
Benin	BJ	616
Bermuda	BM	350
Bhutan	BT	
Bolivia	BO	736
Bosnia and Herzegowina	BA	
Botswana	BW	652
Bouvet Island	BV	
Brazil	BR	724
British Indian Ocean Territory	IO	
Brunei Darussalam	BN	528
Bulgaria	BG	284
Burkina Faso	BF	613
Burundi	BI	642
Cambodia	KH	456
Cameroon	CM	624
Canada	CA	302 and 303
Cape Verde	CV	625
Cayman Islands	KY	346
Central African Republic	CF	623
Chad	TD	622
Chile	CL	730
China	CN	460
Christmas Island	CX	
Cocos (Keeling) Islands	CC	
Colombia	CO	732
Comoros	KM	654
Congo	CG	629

(continued on next page)

Table D–1 (Cont.) Country Codes

COUNTRY	CODES	
	Alphabetic	**Numeric**
Cook Islands	CK	548
Costa Rica	CR	712
Côte d'Ivoire	CI	612
Croatia	HR	
Cuba	CU	368
Cyprus	CY	280
Czech Republic	CZ	230
Denmark	DK	238
Djibouti	DJ	638
Dominica	DM	366
Dominican Republic	DO	370
East Timor	TP	
Ecuador	EC	740
Egypt	EG	602
El Salvador	SV	706
Equatorial Guinea	GQ	627
Eritrea	ER	
Estonia	EE	
Ethiopia	ET	636
Falkland Islands (Malvinas)	FK	
Faroe Islands	FO	288
Fiji	FJ	542
Finland	FI	244
France	FR	208 and 209
France, Metropolitan	FX	
French Antilles		340
French Guiana	GF	742
French Polynesia	PF	547
French Southern Territories	TF	
Gabon	GA	628
Gambia	GM	607
Georgia	GE	
Germany	DE	262 to 265

(continued on next page)

Table D–1 (Cont.) Country Codes

COUNTRY	CODES	
	Alphabetic	**Numeric**
Ghana	GH	620
Gibraltar	GI	266
Greece	GR	202
Greenland	GL	290
Grenada	GD	352
Guadeloupe	GP	
Guam	GU	535
Guatemala	GT	704
Guinea	GN	611
Guinea-Bissau	GW	632
Guyana	GY	738
Haiti	HT	372
Heard and McDonald Islands	HM	
Honduras	HN	708
Hong Kong	HK	453 and 454
Hungary	HU	216
Iceland	IS	274
India	IN	404
Indonesia	ID	510
Iran	IR	432
Iraq	IQ	418
Ireland	IE	272
Israel	IL	425
Italy	IT	222
Ivory Coast see Côte D'Ivoire		
Jamaica	JM	338
Japan	JP	440 to 443
Jordan	JO	416
Kazakhstan	KZ	
Kenya	KE	639
Kiribati	KI	545
Korea, Democratic People's Republic of	KP	467
Korea, Republic of	KR	450, 480, 481

(continued on next page)

Table D–1 (Cont.) Country Codes

COUNTRY	CODES	
	Alphabetic	Numeric
Kuwait	KW	419
Kyrgyzstan	KG	
Lao People's Democratic Republic	LA	457
Latvia	LV	
Lebanon	LB	415
Lesotho	LS	651
Liberia	LR	618
Libyan Arab Jamahiriya	LY	606
Liechtenstein	LI	
Lithuania	LT	
Luxembourg	LU	270
Macau	MO	455
Macedonia, the Former Yugoslav Republic of	MK	
Madagascar	MG	646
Malawi	MW	650
Malaysia	MY	502
Maldives	MV	472
Mali	ML	610
Malta	MT	278
Marshall Islands	MH	
Martinique	MQ	
Mauritania	MR	609
Mauritius	MU	617
Mayotte	YT	
Mexico	MX	334
Micronesia	FM	550
Moldova, Republic of	MD	
Monaco	MC	212
Mongolia	MN	428
Montserrat	MS	354
Morocco	MA	604
Mozambique	MZ	643
Myanmar	MM	414
Namibia	NA	649

(continued on next page)

Table D–1 (Cont.) Country Codes

COUNTRY	CODES	
	Alphabetic	**Numeric**
Nauru	NR	536
Nepal	NP	429
Netherlands	NL	204
Netherlands Antilles	AN	362
Neutral Zone	NT	
New Caledonia	NC	546
New Zealand	NZ	530
Nicaragua	NI	710
Niger	NE	614
Nigeria	NG	621
Niue	NU	
Norfolk Island	NF	
Northern Marianas	MP	534
Norway	NO	242
Oman	OM	422
Pakistan	PK	410
Palau	PW	
Panama	PA	714
Papua New Guinea	PG	537
Paraguay	PY	744
Peru	PE	716
Philippines	PH	515
Pitcairn	PN	
Poland	PL	260
Portugal	PT	268
Puerto Rico	PR	330
Qatar	QA	427
Réunion	RE	647
Romania	RO	226
Russian Federation	RU	250
Rwanda	RW	635
St. Helena	SH	

(continued on next page)

Table D–1 (Cont.) Country Codes

COUNTRY	CODES	
	Alphabetic	Numeric
St. Kitts and Nevis	KN	356
Saint Lucia	LC	358
St. Pierre and Miquelon	PM	308
Saint Vincent and the Grenadines	VC	360
Samoa, Western	WS	549
San Marino	SM	292
Sao Tome and Principe	ST	626
Saudi Arabia	SA	420
Senegal	SN	608
Seychelles	SC	633
Sierra Leone	SL	619
Singapore	SG	525
Slovakia	SK	
Slovenia	SI	
Solomon Islands	SB	540
Somalia	SO	637
South Africa	ZA	655
South Georgia and the South Sandwich Islands	GS	
Spain	ES	214
Sri Lanka	LK	413
Sudan	SD	634
Surinam	SR	746
Svalbard and Jan Mayen Islands	SJ	
Swaziland	SZ	653
Sweden	SE	240
Switzerland	CH	228
Syrian Arab Republic	SY	417
Taiwan, China	TW	466
Tajikistan	TJ	
Tanzania, United Republic of	TZ	640
Thailand	TH	520
Togo	TG	615
Tokelau	TK	
Tonga	TO	539
Trinidad and Tobago	TT	374
Tunisia	TN	605

(continued on next page)

Table D–1 (Cont.) Country Codes

COUNTRY	CODES	
	Alphabetic	**Numeric**
Turkey	TR	286
Turkmenistan	TM	
Turks and Caicos Islands	TC	376
Tuvalu	TV	
Uganda	UG	641
Ukraine	UA	255
United Arab Emirates	AE	424
United Arab Emirates (Abu Dhabi)		430
United Arab Emirates (Dubai)		431
United Kingdom	GB	234 to 237
United States	US	310 to 316
United States Minor Outlying Islands	UM	
Uruguay	UY	748
Uzbekistan	UZ	
Vanuatu	VU	541
Vatican City State (Holy See)	VA	225
Venezuela	VE	734
Viet Nam	VN	452
Virgin Islands (British)	VG	348
Virgin Islands (U.S.)	VI	332
Wallis and Futuna Islands	WF	543
Western Sahara	EH	
Yemen, Republic of	YE	421
Yugoslavia	YU	220
Zaire	ZR	630
Zambia	ZM	645
Zimbabwe	ZW	648

Glossary

This glossary contains brief explanations of important terms used in the MAILbus 400 MTA documentation set. It explains terms as they are used in the context of the MAILbus 400 MTA.

A list of abbreviations is provided in the Preface.

Refer to the HP Enterprise Directory Service documentation set for explanations of the following X.500 terminology used in the MAILbus 400 MTA documentation set:

access control
directory user agent (DUA)
directory system agent (DSA)
distinguished name
distinguished value
master copies
master DSA
naming context
relative distinguished name (RDN)
replication
shadow copies
shadow DSA

1984 downgrading

The process by which an MTA prepares a message for transfer to a messaging system that conforms to the 1984 MHS Standards and that uses the application context MTS transfer protocol 1984. *See also* **application context**.

1984 MHS Standards

The CCITT 1984 X.400 Recommendations concerning electronic message transfer. *See also* **International Telegraph and Telephone Consultative Committee (CCITT)**, **X.400**.

1988 MHS Standards

The CCITT 1988 X.400 Series of Recommendations and International Standard ISO/IEC 10021 concerning electronic message transfer. *See also* **International Organization for Standardization (ISO)**, **International Telegraph and Telephone Consultative Committee (CCITT)**, **X.400**.

1992 MHS Standards

A term used to refer to the 1988 MHS Standards and the following revision documents:

- Revision to the CCITT 1988 X.400 Series of Recommendations:

  The MHS Implementor's Guide Version 10 of February 1993
- Revisions to individual parts of International Standard ISO/IEC 10021:
  - Part 10021-1: Corrigenda 1, 2, 3, 4, 5, 6, and Amendment 2
  - Part 10021-2: Corrigenda 1, 2, 3, 4, 5, 6, 7, and Amendments 1, 2
  - Part 10021-4: Corrigenda 1, 2, 3, 4, 5, 6, 7, 8, and Amendment 1
  - Part 10021-5: Corrigenda 1, 2, 3, 4, 5, 6, 7
  - Part 10021-6: Corrigenda 1, 2, 3, 4, 5, 6, 7
  - Part 10021-7: Corrigenda 1, 2, 3, 4, 5, and Amendment 1

In the MAILbus 400 MTA documentation, any reference to the 1992 MHS Standards also includes the 1988 MHS Standards, unless specifically indicated otherwise.

See also **International Organization for Standardization (ISO)**, **International Telegraph and Telephone Consultative Committee (CCITT)**, **X.400**.

Abstract Syntax Notation One (ASN.1)

An international standard notation used to describe data. ASN.1 notation can be used to define the abstract syntax of information. It is defined in International Standards ISO 8824 and ISO 8825 as well as CCITT Recommendations X.208 and X.209.

Access Unit

An application that links an X.400 MTS to another communication system, for example, a telex network.

Access Units in an MTS based on the MAILbus 400 MTA are referred to as Gateways.

Accounting

An MTA feature that is used to collect specific items of information about the messages an MTA exchanges with its User Agents, Gateways, Message Stores and peer MTAs in other routing domains. The type of information that is collected is determined by characteristic attributes of the MTA, called Accounting filters.

ACSE

See **Association Control Service Element (ACSE)**.

activity

The transfer of a single MPDU from an MTA to another MTA over an association. *See also* **association**.

Activity entity

An entity of the MTA module that holds information about an association and the transfer of an MPDU over that association.

ADMD

See **administration management domain (ADMD)**.

administration management domain (ADMD)

A collection of messaging components, such as MTAs, User Agents and Gateways normally operated by public authorities, such as a Post, Telegraph and Telephone authority. An ADMD provides a commercial message transfer service to its customers, mainly private management domains (PRMDs). An ADMD is confined to one country, but a country can have more than one ADMD. ADMDs are interconnected on a national and international level, thus enabling users of different ADMDs to exchange messages. *See also* **private management domain (PRMD)**.

Agent

A collective term for the components of an MHS that link other messaging systems or users to an MTS. User Agents, Gateways and Message Stores are examples of such components. *See also* **registered Agent**, **unregistered Agent**.

Agent entity

An entity of the MTA module that provides management of the communication between an MTA and a registered Agent. *See also* **registered Agent**.

API

See **MAILbus 400 Application Program Interface (API)**.

application context

Information agreed through negotiation between two MTAs during the establishment of an association. The application context defines the protocols used in the transfer of messages between MTAs. The application contexts are:

- MTS transfer, which indicates that message transfer between a MAILbus 400 MTA and a peer MTA takes place over OSI ACSE connections.
- MTS transfer protocol, which indicates that a peer MTA of the MAILbus 400 MTA accepts 1992 X.400 messages using OSI Session connections.
- MTS transfer protocol 1984, which indicates that a peer MTA of the MAILbus 400 MTA only accepts 1984 X.400 messages using OSI Session connections. The boundary MAILbus 400 MTA must downgrade a message before transferring it to the other routing domain.

See also **1984 downgrading**, **association**.

Archiving

An MTA feature that can be used to keep disk copies of the messages that an MTA exchanges with particular registered Agents and peer MTAs in other routing domains.

area

A geographical location that contains a group of MTAs. If area routing is used, messages that are addressed to MTAs in the area are handled by one MTA or MTA set, known as the area server. *See also* **Area entity**, **area routing**.

Area entity

An entity of the MTS module that represents information in the directory about an area and, if area routing is used, about the area's server. The name of an Area entity is part of the name of each MTA that is located within the area represented by the entity. *See also* **area**, **area routing**.

area routing

The process of transferring messages to an area through an MTA or MTA set which is nominated as the area server. The area server splits each message locally into the required number of MPDUs.

Area routing is used to reduce message traffic over expensive communication lines (for example, X.25 lines) or over communication lines that tend to be congested. *See also* **area**, **message splitting**.

ASN.1

See **Abstract Syntax Notation One (ASN.1)**.

association

A connection between two entities in Application layers on different nodes.

Associations are created between MTAs in order to exchange messages. *See also* **inbound association**, **OSI Reference Model**, **outbound association**.

Association Control Service Element (ACSE)

A protocol in the Application layer that controls how associations between two OSI applications are established and released.

attribute

(1) A component of an O/R address. An O/R address consists of a set of attributes, their types and their values.

(2) An item of information that forms part of an EMA entity. Entities can have identifier, characteristic, status and counter attributes.

(3) A component of an entry in the directory. Each directory entry consists of a set of attributes, their types and their values.

bad message

A message, report or probe that could not be routed by an MTA. The MTA copies a bad message to the bad messages directory.

body

See **IPM body**.

bodypart

See **IPM bodypart**.

Bodypart entity

An entity of the MTA module that holds information about an IPM bodypart. A Bodypart entity is required for each IPM bodypart that the MTA is to convert. *See also* **conversion**, **Converter entity**.

boundary MTA

An MTA within a routing domain that exchanges messages with one or more peer MTAs in one or more other X.400 routing domains. *See also* **peer MTA**.

CCITT

See **International Telephone and Telegraph Consultative Committee (CCITT)**.

CDA

See **Compound Document Architecture (CDA)**.

characteristic attribute

Within EMA, an entity attribute with a value that can be modified using management commands. Changes to characteristic attributes influence the behavior of an entity. A characteristic attribute cannot be modified as the result of an action by an entity.

Changes to the characteristic attributes of the entities in the MTS module have a different effect. Changing a characteristic attribute of an entity in the MTS module means changing information held in the directory.

CLNS

See **connectionless network service (CLNS)**.

complete O/R address

An O/R address that, in terms of the O/R address attribute hierarchy, contains an O/R address attribute that typically identifies a particular MHS user. The following list shows examples of complete O/R addresses:

- A complete mnemonic O/R address (1992 MHS Standards):
  C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=WELL;CN=Tammy Simpson
- A complete mnemonic O/R address containing a Personal Name (1984 MHS Standards):
  C=NZ;A=NZ-PTT;P=MACE;O=MACE;G=Hans;S=Fiedler
- A complete numeric O/R address:
  C=NZ;A=NZ-PTT;N-ID=7483009

See also **partial O/R address**.

compound document

A document that can contain different types of data, such as text, geometric graphics and raster images.

Compound Document Architecture (CDA)

A HP-defined, coordinated set of architectures and services that enable the creation, display, printing, filing, retrieval and distribution of compound documents. CDA allows compound documents to be easily interchanged between conforming applications in a network. CDA stores documents in the DIGITAL Document Interchange Format (DDIF) and the DIGITAL Tabular Interchange Format (DTIF). *See also* **compound document**, **DEC ODA Compound Document Architecture (CDA) Gateway**.

connectionless network service (CLNS)

A network service that operates according to a datagram model. This network service is based on the idea that data is independent and self-contained. For example, individual portions of data from one MPDU are independently routed and delivered to the MPDU's destination. This type of network service is sometimes called connectionless-mode network service.

connectionless-mode network service

See **connectionless network service (CLNS)**.

connection-mode network service

See **connection-oriented network service (CONS)**.

connection-oriented network service (CONS)

A network service in which a logical connection between the source and destination must be established before data can be exchanged. Service data units sent over the connection do not have to contain a destination address. This type of network service is sometimes called connection-mode network service.

CONS

See **connection-oriented network service (CONS)**.

content

The part of a message that contains the information to be transferred to the recipient. In an IPM, the content consists of a heading and a body.

content type

The type of data in a message. The content type is indicated in the message envelope and defines the syntax and the semantics of the data in the content. *See also* **content**.

conversion

A process by which an MTA converts an IPM bodypart format to another IPM bodypart format according to information stored with the recipient's O/R address entry in the directory. *See also* **Converter entity**, **IPM bodypart**, **IPM bodypart converter**.

converter

See **IPM bodypart converter**.

Converter entity

An entity of the MTA module that holds information about a specific IPM bodypart converter that the MTA can use. *See also* **conversion**.

counter attribute

Within EMA, an entity attribute that contains statistical information for that entity. For example, it records the number of times a particular event has occurred. The entities of the MTS module do not have counter attributes.

DEC ODA Compound Document Architecture (CDA) Gateway

The HP software used by the MAILbus 400 MTA for OpenVMS to convert between IPM bodyparts that contain documents in the office document architecture format and HP's compound document architecture format. *See also* **Office Document Architecture (ODA)**.

DECnet-Plus

The HP hardware and software that implements the DIGITAL Network Architecture (DNA) and supports the integration of Open Systems Interconnection (OSI). *See also* **Open Systems Interconnection (OSI)**.

Deferred Message entity

An entity of the MTA module that holds information about a deferred message. A deferred message is a message whose processing has been delayed for a specific period of time, at the request of the originator.

delivery

The process by which an MTA passes a message to a User Agent or Message Store.

delivery report

An MPDU indicating the successful delivery of a message to a recipient. A delivery report is created by an MTA and is sent to the originator's User Agent at the originator's request. *See also* **non-delivery report**.

Dialogue Mode

The mode by which messages are exchanged across associations. The mode can be either monologue or two-way alternate. In monologue dialogue mode, the MTA uses a single association to either send or receive messages, but not both at the same time. In two-way alternate dialogue mode, the MTA uses a single association to alternately send and receive messages

HP Enterprise Directory Service

See **directory**.

director

An EMA management component that interacts with a user, initiates management operations on behalf of the user, coordinates management activities with entities, and provides high-level management access. Network Control Language (NCL) is an example of a director.

directory

A shared, hierarchical database used by the MTA to store O/R addresses and routing information. The directory used by the MAILbus 400 MTA is the HP Enterprise Directory Service, which is based on the 1993 edition of the X.500 Series of Recommendations and International Standard ISO/IEC 9594. The information in the directory is maintained using entities of the MTS module. *See also* **MTS module**, **X.500**.

directory entry

An item of information held in the directory, for example, information about another routing domain to which your routing domain can connect.

Directory name

An identifier used to identify a user in a messaging system. A Directory name can be part of an O/R name.

The MAILbus 400 MTA does not route messages that are addressed with O/R names that only contain Directory names. *See also* **O/R name**.

distribution list

A way of specifying a number of message recipients using a single O/R name. A distribution list is identified by an O/R name and can contain the O/R names of recipients and of other (nested) distribution lists. When a message is addressed to a distribution list, the MTA expands the distribution list and sends an MPDU to each recipient named in the distribution list and in any nested distribution lists.

The MAILbus 400 MTA only routes O/R names in a distribution list that contain O/R addresses. *See also* **O/R address**, **O/R name**.

domain

See **routing domain**, **X.400 management domain**.

Domain entity

An entity of the MTS module that represents information about a routing domain and indicates how messages are routed to recipients within that routing domain. There is a Domain entity for each routing domain that is directly or indirectly connected to your routing domain. *See also* **routing domain**.

downgrading

See **1984 downgrading**.

EDI

See **electronic data interchange (EDI)**.

EIT

See **encoded information type (EIT)**.

electronic data interchange (EDI)

The exchange of business data between computers in standardized, structured formats.

EMA

See **Enterprise Management Architecture (EMA)**.

encoded information type (EIT)

An identifier or set of identifiers that describe a particular data format within a message content, for example, an IPM bodypart. EITs are listed in the message envelope to indicate the presence of a particular data format within the message content. EITs can also be stored as part of O/R address entries in the directory to indicate which data formats can be delivered to recipients at these O/R addresses.

Enterprise Management Architecture (EMA)

The HP model of a system that allows the management of an entire distributed computer environment. EMA describes concepts and rules which ensure consistent management of applications and systems software in a network. *See also* **director**, **entity**, **event**.

entity

An individually manageable part of an EMA network. An entity has attributes which identify and describe it. You can change the behavior of an entity by using commands to modify some of the entity's attributes.

The entities of the MTS module are different from other EMA entities. When you change the attributes of the entities in the MTS module, you modify the information in the directory that these entities represent. *See also* **attribute**.

entry

See **directory entry**.

envelope

The part of a message that contains the information an MTS requires to transfer a message from the originator's User Agent to the recipient's User Agent.

event

An occurrence of a normal or abnormal condition detected within an entity and of interest to EMA network management. *See also* **entity**, **event sink**.

event sink

The destination of entity events. For example, an event sink can be a file or a display on the operator's console. *See also* **entity**, **event**.

export

The process by which an MTA passes a message to a Gateway.

extensions

The 1992 Elements of Service defined by CCITT and ISO in addition to the 1984 Elements of Service. Elements of Service define features, functions or capabilities of an X.400 MHS. The Elements of Service defined for a 1992 X.400 MHS are listed in the 1992 MHS Standards.

Extensions are removed by the MTA when preparing a message for a 1984 routing domain in the process of 1984 downgrading.

Foreign Address entity

An entity of the MTS module that represents information about a mail address format valid in another routing domain and an equivalent O/R address valid in your routing domain. This information is used for address translation at Gateways. *See also* **Gateway**.

Gateway

An application that links a routing domain based on the MAILbus 400 MTA to another messaging system, typically a non-X.400 messaging system. In the 1992 MHS Standards, Gateways are referred to as Access Units.

GDI

See **global domain identifier (GDI)**.

global domain identifier (GDI)

The set of O/R address attributes that identifies a routing domain and that is part of the O/R addresses of all users in that routing domain. GDIs are assigned by one or more registration authorities. A GDI comprises attribute values for the country and the administration management domain, and optionally, the private management domain.

global entity

A member of the highest level entity class within EMA. For the MTA and MTS modules the global entity is NODE.

heading

Part of an IPM content. The heading contains control information that is used by User Agents.

identifier attribute

Within EMA, an attribute that identifies an entity for the purpose of network management.

import

The process by which an MTA receives a message from a Gateway.

inbound association

An association that is initiated by a peer MTA and accepted or rejected by an MTA for the transfer of a message from the peer MTA to the accepting MTA.

inbound message

A message coming into an MTA either from a peer MTA through the Relayer, or from an Agent through the Interface Region.

Interface Region

A functional part of the MTA that is responsible for interacting with Agents served by the MTA.

International Organization for Standardization (ISO)

An international standards body responsible for the OSI network standards and the OSI Reference Model. *See also* **1992 MHS Standards**, **OSI Reference Model**, **Open Systems Interconnection (OSI)**.

International Telegraph and Telephone Consultative Committee (CCITT)

The technical committee of the International Telecommunications Union (ITU), responsible for the development of recommendations regarding telecommunications, including data communications. The abbreviation CCITT is derived from the French name of the committee: Comité Consultatif International Télégraphique et Téléphonique. The CCITT Recommendations include the X.400 Series of Recommendations for Message Handling Systems.

The CCITT is now the ITU–T (International Telephone Union – Telecommunications). Their published documents still have CCITT identification material, and to avoid confusion this book still uses the term CCITT.

See also **1984 MHS Standards**, **1992 MHS Standards**, **X.400**.

interpersonal message (IPM)

A message exchanged between people (users) in an electronic messaging system.

IPM

See **interpersonal message (IPM)**.

IPM body

The main part of an IPM content that contains the information the user wants to communicate. The body consists of a sequence of one or more IPM bodyparts and does not contain addressing or routing information. *See also* **IPM bodypart**.

IPM bodypart

An element of an IPM that contains a piece of information that the user wants to communicate. An IPM can contain several different IPM bodyparts, for example, text and graphics, and an IPM bodypart can itself be an IPM containing IPM bodyparts. *See also* **IPM bodypart converter**.

IPM bodypart converter

A software tool that transforms data from one encoding or data format to another. An IPM bodypart converter can be used by an MTA to convert an IPM bodypart that is unacceptable to a recipient's Agent to an IPM bodypart that the Agent can accept. *See also* **conversion**, **IPM bodypart**.

ISO

See **International Organization for Standardization (ISO)**.

journal

A record of MPDUs that the MTA has successfully routed to an Agent or another MTA and no longer has responsibility for. In the event of a system or MTA shutdown and recovery, the MTA uses the journal in conjunction with the messages in the stable workspace to determine the recipients for which a message must be recovered. The keeping of a journal prevents an MTA from sending multiple copies of a message to a recipient.

mailbox

A delivery queue used by the MTA when delivering messages to an unregistered Agent.

MAILbus

A term registerd by HP used to refer to a set of software products that provides the backbone for an electronic messaging system. MAILbus is a collection of products; it is not itself a product. MAILbus products are based on standards that have since been superseded. Examples of MAILbus products are Message Router and Message Router X.400 Gateway (MRX).

MAILbus 400

A term registered by HP used to refer to the set of software products that conform to the OSI and X.400 standards and that provide the backbone for an electronic messaging system in an open network environment. MAILbus 400 is a collection of products, it is not itself a product. Examples of MAILbus 400 products are the MAILbus 400 MTA and the HP MAILbus 400 SMTP Gateway.

MAILbus 400 Application Program Interface (API)

A callable interface that is used to build applications that access MAILbus 400 MTAs and use their message transfer service. The MAILbus 400 API conforms to the X/Open Common Applications Environment (CAE) specification *API to Electronic Mail (X.400)*.

MAILbus 400 Message Router Gateway

A Gateway that can be used to connect a messaging system based on the MAILbus 400 MTA to a Message Router system.

MAILbus 400 Message Store

HP's implementation of a Message Store that conforms to the CCITT Recommendation X.413 and International Standard ISO/IEC 10021-5.

MAILbus 400 MTA

HP's implementation of a Message Transfer Agent that conforms to the 1992 MHS Standards, which include the CCITT 1988 X.400 Series of Recommendations and International Standard ISO/IEC 10021, and a set of revision documents published since these standards and recommendations were issued. The MAILbus 400 MTA is the core component of the MAILbus 400 product set, and creates the basis of an open, standards-based messaging system. *See also* **1992 MHS Standards**.

HP MAILbus 400 SMTP Gateway

A Gateway that can be used in conjunction with the MAILbus 400 MTA. It provides connectivity to a UNIX-based sendmail system in an internet network. This provides access to any of sendmail's local mailers, such as UUCP, MAIL-11 or any remote application that uses the Simple Mail Transfer Protocol (SMTP).

management domain

See **X.400 management domain**.

message

The information to be transmitted in an MTS from the originator to the recipient(s). A message consists of an envelope and a content.

Message Handling System (MHS)

A network of computer systems that handles messages. A Message Handling System contains at least one MTA and one User Agent.

Message History logging

An MTA feature that can be used to collect information about the messages that an MTA handles. The resulting information allows you to trace messages in the routing domain. *See also* **Processed Message entity**.

Message Processor

A functional part of the MTA that is responsible for processing messages. The Message Processor performs actions such as routing, message splitting, conversion, area routing, distribution list expansion, redirection, and report generation.

Message Protocol Data Unit (MPDU)

An instance of a message, probe or report in an MTS. Multiple MPDUs of a message or probe are created in an MTS through message splitting. *See also* **message splitting**, **MPDU entity**.

message splitting

A process by which MTAs replicate a message or probe. During message splitting, MTAs create different instances of a message or probe, as many as are required to deliver the message or probe to each intended destination. *See also* **Message Protocol Data Unit (MPDU)**.

Message Store

An application that provides message storage and submission services for the users in an X.400 MHS.

Message Transfer Agent (MTA)

A functional component within an MHS that relays messages between Agents. A number of MTAs constitute an MTS. *See also* **Message Transfer System (MTS)**.

Message Transfer System (MTS)

A collection of MTAs that provide a store-and-forward message transfer service. The message transfer service provided by an MTS is used by applications such as User Agents and Gateways. *See also* **Message Handling System (MHS)**.

MHS

See **Message Handling System (MHS)**.

mnemonic O/R address

An O/R address that mnemonically identifies a user or distribution list. A term is mnemonic if it is intended to be easily remembered. A mnemonic O/R address consists of a hierarchical sequence of names, logical abbreviations or acronyms.

module

A group of network functions within EMA which together provide a particular service. A module consists of entities which enable the management of the module's functions. *See also* **MTA module**, **MTS module**.

MPDU

See **Message Protocol Data Unit (MPDU)**.

MPDU entity

An entity of the MTA module that holds information about an MPDU and its progress through an MTA. *See also* **Message Protocol Data Unit (MPDU)**.

MTA

See **Message Transfer Agent (MTA)**.

MTA entity

(1) The top level entity of the MTA module that allows management of an MTA.

(2) An entity of the MTS module that represents information in the directory about an MTA in the routing domain.

There is an MTA entity of the MTA module and an MTA entity of the MTS module for each MTA in a routing domain.

MTA module

The module whose entities allow management of the MAILbus 400 MTA product and its connections to peer MTAs and Agents.

MTA set

A group of MTAs that are configured to provide equivalent services. MTA sets can be used where a constant availability of MTAs is required or where single points of failure can occur. MTA sets are used, for example, as an area server or as a boundary MTA.

MTS entity

The top level entity of the MTS module. The MTS entity represents information in the directory that is shared by all MTAs in the routing domain, for example, O/R addresses and routing information. *See also* **MTS module**.

MTS module

The module whose entities provide management access to the directory information shared by MAILbus 400 MTAs in a routing domain.

The entities of the MTS module are different from other EMA entities. They do not hold information themselves, but represent the information that is held in the directory. The entities of the MTS module do not have counter or status attributes and do not issue events.

multihomed routing domain

A routing domain that is identified by more than one global domain identifier (GDI). All recipients in a multihomed routing domain can be addressed through any of the global domain identifiers. *See also* **global domain identifier (GDI)**.

NCL

See **Network Control Language (NCL)**.

Network Control Language (NCL)

A command line interface used for network management. NCL replaces the Network Control Program (NCP). NCL is the EMA director you can use to manage the entities of the MTA and MTS modules. *See also* **director**.

non-delivery report

A report that indicates that the MTS was unable to deliver a message to a recipient. A non-delivery report is created by an MTA and, when requested by the originator, is sent to the originator's User Agent. *See also* **delivery report**.

numeric O/R address

An O/R address that uniquely identifies a user in a messaging system by means of a number.

ODA

See **Office Document Architecture (ODA)**.

Office Document Architecture (ODA)

An international standard that defines a structure for compound documents and a way of interchanging them between different vendors' systems. ODA stores documents in the Office Document Interchange Format (ODIF). Both ODA and ODIF are defined in *International Standard ISO 8613, Information Processing - Text and Office Systems - Office Document Architecture (ODA) and Interchange Format*. *See also* **compound document**, **DEC ODA Compound Document Architecture (CDA) Gateway**.

object identifier

A sequence of numbers which uniquely identifies an object. Each number is assigned by a specific authority in a defined hierarchy, thus enabling object identifiers to be globally unique. Object identifiers are used, for example, to identify IPM bodyparts in an IPM.

Open Systems Interconnection (OSI)

A set of international standards developed by the International Organization for Standardization (ISO). The goal of OSI is to allow different vendors' computer systems to be freely interconnected. *See also* **International Organization for Standardization (ISO)**, **OSI Reference Model**.

O/R address

A hierarchical set of attributes that enables MTAs to distinguish one user from another and to deliver messages or reports to these users. An O/R address is part of an O/R name. *See also* **mnemonic O/R address**, **numeric O/R address**, **O/R name**, **postal O/R address**, **terminal O/R address**.

O/R name

Information that identifies a specific user. According to the 1992 MHS Standards, an O/R name can consist of an O/R address and/or a Directory name.

In the context of the MAILbus 400 MTA an O/R name must contain an O/R address, because MAILbus 400 MTAs use O/R addresses for routing. Note that in 1984 messaging systems the term O/R name is used to refer to O/R addresses. *See also* **O/R address**.

ORaddress entity

An entity of the MTS module that represents an O/R address entry in the directory. The ORaddress entity also contains information about how messages are to be routed to this O/R address, the permissible format of these messages, and other information relating to the recipient(s) at the O/R address.

originator

The user or application sending a message. *See also* **recipient**.

OSI

See **Open Systems Interconnection (OSI)**.

OSI Reference Model

The abbreviation for the Reference Model for Open Systems Interconnection. The OSI Reference Model is a seven-layer conceptual model for a network architecture. It defines the International Standards to be used by open systems. Starting from the top, the layers are: Application, Presentation, Session, Transport, Network, Data link, and Physical. Each layer provides services to the layer above and uses the services provided by the layer below. *See also* **International Organization for Standardization (ISO)**.

outbound association

An association initiated by an MTA and accepted or rejected by a peer MTA for the transfer of a message from the initiating MTA to the peer MTA.

outbound message

A message that an MTA passes either to another MTA through the Relayer or to an Agent through the Interface Region.

partial O/R address

An O/R address that, in terms of the hierarchy of O/R address attributes, contains a number of the leading O/R address attributes of an O/R address and identifies a group of users.

The following are examples of partial O/R addresses:

- A partial mnemonic O/R address:
  C=NZ;A=NZ-PTT;P=ACME;O=ACME;OU1=AUCK

- A partial postal O/R address:
  C=NZ;A=NZ-PTT;P=ACME;PD-C=530

See also **complete O/R address**.

peer MTA

An MTA with which an MTA can exchange messages. An MTA can have peer MTAs in the same or in another routing domain. *See also* **Peer MTA entity**.

Peer MTA entity

An entity of the MTA module that holds information that a particular MTA needs to know about a peer MTA to which it can connect.

If an MTA is to exchange messages with a peer MTA in another routing domain, you must manually create a Peer MTA entity to supply information about the peer MTA in the other routing domain. These Peer MTA entities are called manually-configured Peer MTA entities. An MTA can be connected to several peer MTAs in one or more other routing domains, in which case it has several manually-configured Peer MTA entities.

When an MTA exchanges messages with a peer MTA that is in the same routing domain, the MTA automatically creates a Peer MTA entity when it establishes an association with the peer. *See also* **boundary MTA**, **peer MTA**.

postal O/R address

An O/R address that identifies a user by means of the user's postal address. It identifies the physical delivery system through which the user can be accessed and indicates the user's postal address.

Presentation address

The address that an MTA uses to establish a connection to another MTA over the Presentation layer. Presentation addresses are used when two MTAs that conform to the 1992 MHS Standards establish a connection. *See also* **OSI Reference Model**.

private management domain (PRMD)

A messaging domain controlled by an organization other than a CCITT administration, for example, a company. A private management domain does not generally offer commercial network or messaging services to any third party, but simply meets its own messaging needs. *See also* **administration management domain (ADMD)**.

PRMD

See **private management domain (PRMD)**.

probe

An MPDU that consists only of an envelope and that has the characteristics of a message that the originator wishes to send. A probe can help to determine whether any aspect of the message, for example, its size, content type and IPM bodyparts, would prevent the MTS from delivering the message to its intended recipients.

Processed Message entity

An entity of the MTA module that holds information about a message (not about reports and probes). This information is about messages currently being processed at the MTA. If Message History logging is enabled, the Processed Message entity can also contain information about messages the MTA has processed in the past. *See also* **Message History logging**.

reassignment

The process by which an MTA replaces the O/R address of a user or distribution list on a message or probe with an alternative O/R address. The term reassignment is used in the MAILbus 400 Application Program Interface documentation. In the MAILbus 400 MTA documentation this process is referred to as redirection.

recipient

The user or application receiving a message. *See also* **originator**.

recovery

The procedure by which an MTA takes responsibility for the messages in the workspace of a peer MTA in the same routing domain, and attempts to process those messages.

redirection

The process by which an MTA replaces the O/R address of a user or distribution list on a message or probe with an alternative O/R address. The term redirection is used in the MAILbus 400 MTA documentation. In the MAILbus 400 Application Program Interface documentation this process is referred to as reassignment.

registered Agent

An Agent of the MTA that is represented by an Agent entity and that can serve one or more O/R addresses. Typically, Gateways, User Agents serving a user community, and Message Stores are registered Agents. *See also* **Agent entity**, **unregistered Agent**.

registration authority

An individual or group responsible for deciding values of O/R address attributes at a particular level in the O/R address attribute hierarchy. Examples of registration authorities are administration management domains (ADMDs) and private management domains (PRMDs). This term is also used to refer to individuals or groups of individuals responsible for assigning object identifiers.

Relayer

A functional part of the MTA that is responsible for transferring messages between an MTA and its peer MTAs.

Reliable Transfer Service Element (RTSE)

Transmission services provided to OSI applications. The RTSE services are used by the MTA to transfer messages, recover from message transfer failures, and resume the message transfer from the point at which it was interrupted.

report

See **delivery report**, **non-delivery report**.

RFC 1006

A protocol that, when layered on top of TCP/IP, emulates the OSI Transport Service Protocol Class 0.

routing

The process by which an MTA uses the recipient information specified in the message envelope to look in the directory for information on how to transfer the message.

The MAILbus 400 MTA only routes messages that are addressed with O/R names that contain O/R addresses. *See also* **O/R name**.

routing domain

A collection of messaging products that access the same routing information. In the context of the MAILbus 400 MTA, your routing domain is the collection of MAILbus 400 MTAs that share the same routing information in the directory.

Routing domains can consist of X.400 messaging products (like the MAILbus 400 MTA) or non-X.400 messaging products. A non-X.400 routing domain can connect to your routing domain through a Gateway. *See also* **Gateway**.

routing information

In the context of MTA management, the information in the directory that MTAs need in order to transfer messages and deliver them to their intended recipients.

RTSE

See **Reliable Transfer Service Element (RTSE)**.

Session address

The address that an MTA uses to establish a connection to another MTA over the Session layer. Session addresses are used when a connection between a MAILbus 400 MTA and an MTA based on the 1984 MHS Standards is established. *See also* **OSI Reference Model**.

Shared File interface

An Agent interface used by Gateways. It allows an MTA or an Agent to place messages in specific queues at the MTA, in the form of files. The MTA or Agent reads the files, processes them and then deletes them from the queue. *See also* **Agent**.

status attribute

Within EMA, an attribute that reflects the behavior of an entity at some point in time. Status attributes cannot be modified using management commands. The entities of the MTS module do not have status attributes.

store-and-forward

A method of transferring a message through a network. When a message arrives at an MTA it is stored at that MTA until it can be forwarded to the next MTA along the message's route.

submission

The process by which a User Agent or Message Store passes a message to an MTA.

TCP/IP

A non-OSI Transport Service, comprising the Transmission Control Protocol (TCP) and Internet Protocol (IP), defined by the Internet community in RFC 791 and RFC 793. TCP/IP corresponds approximately to the Transport and Network layers of the OSI reference model. If the RFC 1006 protocol is layered on top of TCP/IP, the TCP/IP Transport Service emulates the OSI Transport Service Protocol Class 0.

terminal O/R address

An O/R address that identifies users or applications by means of the network address of their terminals. Terminal O/R addresses are mainly used for message exchange with telex, teletex and facsimile machines.

trace information

Information in a message envelope that identifies each X.400 management domain that has handled the message. Trace information also identifies each MTA that has handled the message within an X.400 management domain.

transfer

The process by which an MTA passes a message to a peer MTA.

unregistered Agent

An Agent of the MTA that serves only one O/R address and identifies itself to its MTA using that O/R address. An MTA has less information about its unregistered Agents than about its registered Agents. This is because unregistered Agents are not represented by Agent entities. *See also* **registered Agent**.

User Agent

An application that enables a user to handle messages. A User Agent is software code that interacts with the MTS to send and receive messages. *See also* **Agent**.

workspace

The area used by the MTA when processing messages.

X.25

The set of CCITT Recommendations for the transfer of data over a Packet Switching Data Network. The X.25 Recommendations define the control mechanisms used in the interface between a packet network and a user's terminal. The X.25 Recommendations are published by the CCITT in *Recommendation X.25 Interface Between Data Terminal Equipment and Data Circuit Terminating Equipment for Terminals Operating in the Packet Mode on Public Data Networks*.

X.400

The set of CCITT recommendations and ISO/IEC standards that define the technical basis for an electronic mail system. The X.400 standards and recommendations describe the system model and service elements of an MHS in accordance with the OSI Reference Model.

The 1988 X.400 Series of Recommendations is published by the CCITT in *CCITT Blue Book Volume VIII Fascicle VIII.7, Data Communication Networks - Message Handling Systems (Recommendations X.400-X.420)*. The corresponding ISO Standard ISO/IEC 10021 is published in *International Standard ISO/IEC 10021, Information Processing Systems - Text Communications - Message-Oriented Text Interchange Systems (MOTIS)*.

Since 1988, the CCITT and ISO/IEC have continually revised their work. In 1992, the CCITT have consolidated their revisions with the 1988 X.400 Series of Recommendations to form the 1992 editions of these recommendations. For the purpose of the MTA documentation, the 1988 CCITT X.400 Recommendations, International Standard ISO/IEC 10021, *and* their revisions are collectively called the 1992 MHS Standards. *See also* **1992 MHS Standards**, **International Organization for Standardization (ISO)**, **International Telephone and Telegraph Consultative Committee (CCITT)**, **OSI Reference Model**.

X.400 management domain

An MHS or part of an MHS that includes at least one MTA and that is managed by an administration or an organization. *See also* **administration management domain (ADMD)**, **private management domain (PRMD)**.

X.500

The set of CCITT Recommendations and ISO Standards that define the technical basis of a directory service implementation for use by information processing systems within OSI.

The 1988 X.500 Series of Recommendations is published by the CCITT in *CCITT Blue Book Volume VIII Fascicle VIII.8, Data Communication Networks - Directory (Recommendations X.500-X.521)*. The corresponding ISO Standard ISO/IEC 9594 is published in *International Standard ISO/IEC 9594, Information Technology - Open Systems Interconnection - The Directory*. *See also* **directory**.

X/Open

An independent, worldwide, open systems organization supported by most of the world's largest information systems suppliers. X/Open produces standards specifying application program interfaces (APIs) for Open Systems Interconnection. *See also* **MAILbus 400 Application Program Interface (API)**.

XAPI interface

An Application Program Interface used by Gateways and User Agents to exchange messages with an MTA. An XAPI interface conforms to the *X/Open CAE Specification, API to Electronic Mail (X.400)* developed jointly by the X/Open Company Limited and the X.400 API Association. The HP product which implements this specification is the MAILbus 400 Application Program Interface. *See also* **MAILbus 400 Application Program Interface (API)**, **X/Open**.

Index

A

Abbreviations, xvii
Access control
  and the DSA, 6–3
  and the MTS module, 7–3
Access Unit, 2–3
Accounting, 5–30
ACME
  diagram of routing domain layout, 4–14, 4–15
  example MTS script, B–1
  hierarchy diagram, 3–1
  layout of routing domain, 4–5
  recommended hierarchy, 3–5
ADMD, 5–4
  See Administration Domain Name
Administration Domain Name, 5–8, 5–11
  in a GDI, 4–4
  single space value, 5–11
  single zero value, 5–11
Administration Management Domain, 5–4
Agent, 2–1
  connections using the API Server, 11–2
  identifier, 4–26
  interfaces, 4–12
  Invocation Filename, 4–26
  name, 4–12
  password, 4–12, 4–26
  registered, 4–12
  type, 4–12, 4–26
  types, 4–11
  type Shared File 1984
    setup, 11–5
Agent (cont'd)
  type Shared File 1992
    setup, 11–4
  type XAPI
    setup, 11–1
  unregistered, 4–13
Agent entity, 4–12, 7–4
  creating, 10–4
Alias
  attribute values, 5–10
  O/R address, 5–18, 7–8
Alias target, 7–8
API,
  See MAILbus 400 Application Program Interface
API Server, 11–1
Application, 2–1
Application Context
  for Peer MTA, 4–22
  MTS Transfer, 4–18
  MTS Transfer Protocol, 4–18
  MTS Transfer Protocol 1984, 4–18
Application Program Interface, 4–12
Archiving, 5–30
Area, 4–7
  diagram, 4–8
  dividing your routing domain into, 4–7
  hierarchy, 4–8, 8–11, 9–12
  identifier, 4–20
  naming, 4–7
  server, 4–7
  Server MTA, 4–20
  setting up
    on OpenVMS, 9–12
    on Tru64 UNIX, 8–11

Area and MTA entries in the directory
diagram, 7–15
Area entity
attributes, C–2
creating, 8–11, 9–12
example, 7–15
for ACME, 7–15
Association, 2–2
Attribute
of MTS module entities, C–1
of the Area entity, C–2
of the Domain entity, C–4
of the Foreign Address entity, C–5
of the MTA entity (MTS Module), C–2
of the MTS entity, C–1
of the ORaddress entity, C–2
Authorize
routing information, 8–13, 9–15
Automatically-configured Peer MTA entity, 4–17

B

Body
diagram, 2–6
of IPM, 2–5
Bodypart, 2–5
diagram, 2–6
Boundary MTA
diagram, 4–14
Peer MTA entity, 4–16, 4–17, 5–25

C

CCITT, 1–2
X.121, D–1
CLNS, 1–6
Cluster
and MTA setup, 9–2
Common Name, 5–8, 5–14
Complete O/R address, 5–17
routing information, 5–19
Conformance, 1–2
Connecting to more than one ADMD, 5–11
Connectionless network service, 1–6
Connection-oriented network service, 1–6
CONS, 1–6
Content, 2–4
of IPM, 2–5
Content Information, 5–27
forms of, 5–27
Conventions, xv
Conversion, 2–6, 5–30
Country codes, 5–10
CCITT, 5–10
ISO, 5–10
list of, D–1
Country Name, 5–8
in a GDI, 4–4
Country Name attribute, 5–10
Create
Agent entities, 10–4
Area entity, 8–11, 9–12
GDI attributes in directory, 8–10, 9–11
MTA entity, 8–6, 8–15, 9–6, 9–18
MTS entity in directory on OpenVMS, 9–11
MTS entity in directory on Tru64 UNIX, 8–9
Peer MTA entities, 10–4
routing information for ACME example, B–1
create_mta_cons_template.ncl, 8–7, 8–16

D

DDA, 5–14
Defining a hierarchy, 3–1
Definitive O/R Address
routing instruction, 5–20, 7–11
Deliver
routing instruction, 5–19, 5–23, 5–25, 7–10
Delivery, 2–3
Delivery report, 2–6
Dialogue mode
for Peer MTA, 4–22
for Peer MTA entity, 4–18

Different CCITT Domain, 4–16, 7–12
Director, 1–9
Directory
  distributed, 1–3
  entering information, 1–8
  for storing routing information, 5–6
  lookup, 1–3
    diagram, 1–4
  regulating access to, 6–2
  standards, 1–3
Directory Information Tree, 4–3
Directory name, 5–1
  in distribution list, 5–27
Directory system agent, 1–3
Directory user agent, 1–3
Distinguished name, 4–3
  for routing domain, 4–4
Distributed directory, 1–3
Distributed management, 1–7
Distribution list, 5–27
  diagram, 5–28
  nested, 5–27
Documentation
  for MAILbus 400 MTA, xiii
Domain
  Administration Management, 5–4
  Private Management, 5–4
  routing, 4–1
  X.400 management, 5–4
Domain and Foreign Address entries
  diagram, 7–17
Domain Defined attribute, 5–8, 5–14
Domain entity, 4–15, 7–5
  attributes, C–4
  for ACME, 7–11
Domain entries in the directory, 4–15, 7–11
Domain name
  for Peer MTA entity, 4–19
Domain routing information, 5–25
Downgrade, 5–30
DSA, 1–3
  and MTA setup, 8–2, 9–3
  master copy, 6–1
  shadow copy, 6–1
  updating, 6–4
DSA entity response
  No Parent, 8–8, 9–9
dsa_configure
  on Tru64 UNIX, 8–7
DSA_CONFIGURE
  on OpenVMS, 9–8
DUA, 1–3
  and MTA setup, 8–3, 9–3
dua.defaults, 8–9, 8–12, 8–17, 8–20
DUA defaults
  for an upgrade
    on OpenVMS, 9–4
    on Tru64 UNIX, 8–4
  when adding more MTAs
    on OpenVMS, 9–13
    on Tru64 UNIX, 8–12
dua_configure, 8–9, 8–12, 8–17, 8–20
DXD$DUA_CONFIGURE, 9–10, 9–13, 9–21, 9–23
DXD$DUA_DEFAULTS.DAT, 9–10, 9–14, 9–21, 9–24

E

EDI, 2–5
Electronic data interchange, 2–5
EMA, 1–6
Enterprise Management Architecture, 1–6
Envelope, 2–4
Errors reported by MTAmail, 12–6
Export, 2–3

F

Foreign address entity
  example, 7–18
Foreign Address entity
  attributes, C–5
  for ACME, 7–16
Foreign Address entry in the directory
  diagram, 7–17

G

Gateway, 1–1
  developed for ISOPLEX 800 MTA, 11–4
  developed for OpenServer 400, 11–5
  diagram, 2–2
  MAILbus 400 Message Router Gateway, 1–5
  role of, 2–3
gateway.dat, 11–6, 11–7
GDI attributes
  creating, 8–10, 9–11
Generation, 5–8
Geographical divisions, 3–1
Geographical hierarchy
  diagram, 3–2
Given Name, 5–8
Global Domain Identifier, 4–4, 5–10, 7–6
  for other routing domains, 4–13
  for routing domain, 4–20
  multi-homed, 4–4
  multiple, 5–18

H

Heading
  diagram, 2–6
  of IPM, 2–5
Heading field, 2–5
  diagram, 2–6
Hierarchy
  recommended
    diagram, 3–5
HP DECnet-Plus, 1–6
HP Enterprise Directory Service, 6–1
  See also Directory
  Naming Context entities, 6–2
  object classes, 4–3
  setup requirements for the MTA, 6–2
  Subordinate Reference entities, 6–2

I

Identifier
  for Agent, 4–26
  for Area, 4–20
  for MTA, 4–20
  for MTA Set, 4–20
Import, 2–3
Initials, 5–8
Input Queue
  for the MTA on OpenVMS, 11–7
  for the MTA on Tru64 UNIX, 11–6
Installing
  MTA components, 9–3
  MTA subsets, 8–3
International Organization for Standardization, 1–2
International Telegraph and Telephone Consultative Committee, 1–2
Interpersonal message
  See IPM
Interpersonal Messaging Protocol, 2–5
Interworking, 1–2
  with 1984 systems, 5–1
Invocation Filename
  for Agent, 4–26
IPM, 2–5
  bodypart, 2–5
    diagram, 2–6
  content, 2–5
    diagram, 2–6
  diagram, 2–6
ISO, 1–2
  standard 3166, D–1
ISOPLEX 800 MTA and the MTA, 11–5

L

Layout of ACME routing domain, 4–5
License
  for the MAILbus 400 MTA, 8–2
Local Name
  for Peer MTA, 4–22

Local Password
for Peer MTA, 4–22
Logging
Accounting information, 5–30
Message History information, 5–30

M

MAILbus, 1–4
connecting to MAILbus 400, 1–5
MAILbus 400, 1–1
Application Program Interface, 4–12
connecting to MAILbus, 1–5
diagram, 1–5
Message Router Gateway, 1–5
MAILbus 400 MTA
conformance, 1–2
MAILbus 400 MTA License, 8–2
Management Domain
Administration, 5–4
Private, 5–4
Manually-configured Peer MTA entity, 4–17
Master DSA, 6–3, 7–2
setting up access on OpenVMS, 9–9
setting up access on Tru64 UNIX, 8–8
May Cross CCITT Boundaries, 7–10
MD Redirect
routing instruction, 5–23, 7–8
Members
for MTA Set, 4–20
Message, 2–4
definition used in the MTA documentation, 2–4
delivery, 2–3
export, 2–3
import, 2–3
recovery, 5–29
structure, 2–4
submission, 2–3
Message handling system
See MHS
Message History logging, 5–30
Message protocol data unit, 4–7
Message Router X.400 Gateway, 1–5
Message Store, 1–1
diagram, 2–2
role of, 2–3
Message transfer agent
See MTA
Message Transfer Protocol, 2–4
Message transfer system
See MTS
Message transfer to other routing domains
direct, 4–14
indirect, 4–15
MHS, 2–1
diagram, 2–2
MHS Standards
1984, 1–2
1988, 1–2
1992, 1–2
MH_RC_NO_SUCH_CLIENT, 12–6
Mnemonic Content Information, 5–27
Mnemonic O/R address, 5–2
attributes, 5–8
diagram, 5–5
Mnemonic Routing Instruction, 5–26
MPDU, 4–7
MRX, 1–5
MTA, 2–2
diagram, 2–2
examples of names, 4–10
identifier, 4–20
Input Queue, 11–5
name as represented in the directory, 4–9
name planning, 4–1
Naming Context
on OpenVMS, 9–9
on Tru64 UNIX, 8–8
Output Queue, 11–5
password, 4–11, 4–20
Presentation address, 4–11
running MTA Setup
on OpenVMS, 9–5, 9–15
on Tru64 UNIX, 8–4, 8–13
Shared File 1984 interface, 11–5
Shared File 1992 interface, 11–5

MTA (cont'd)
starting
on OpenVMS, 9–7
on Tru64 UNIX, 8–6
startup
on OpenVMS, 9–18
on Tru64 UNIX, 8–15
startup script, 8–18, 9–20
verification
on OpenVMS, 9–20
on Tru64 UNIX, 8–16
MTA$CREATE_CLIENT_CLNS_ TEMPLATE.NCL, 11–2
MTA$CREATE_CONS_TEMPLATE.NCL, 9–7, 9–19
MTA$MTS_CREATE_MTA_ENTRY.NCL, 9–5, 9–17
MTA$SERVER_SETUP, 9–5, 9–15
MTA$START.NCL, 9–5, 9–17
MTA and Area entry
diagram, 7–15
MTA components
installing, 9–3
MTA entity
attributes, C–2
creating, 8–6, 8–15, 9–6, 9–18
MTA information
for ACME, 7–15
MTAmail, 12–1
MTAmail errors, 12–6
MTA module, 1–7, 7–4
diagram, 1–7, 7–4
MTA name, 4–10
MTA set, 4–5
as a boundary MTA, 4–5, 4–17
as an area server, 4–5
directory entry, 4–6
example, 7–16
identifier, 4–20
members, 4–20
planning, 4–6
when to use, 4–5
MTA setup
adding more MTAs to your routing domain, 8–11, 9–13
for the first MTA
MTA setup
for the first MTA (cont'd)
on OpenVMS, 9–8
on Tru64 UNIX, 8–7
providing authorization, 8–13, 9–15
when upgrading
on OpenVMS, 9–4
on Tru64 UNIX, 8–3
MTA setup on a cluster, 9–2
MTA subsets
installing, 8–3
mta_api_server_address, 11–2
MTA_NODE, 11–2
mta_setup, 8–4, 8–13
MTS, 2–2
diagram, 2–2
MTS entity
attributes, C–1
creating, 8–9, 9–10
on OpenVMS, 9–11
on Tru64 UNIX, 8–9
name, 4–4
password, 8–9, 9–10
using, 7–2
MTS module, 1–8, 7–1
and access control, 7–3
diagram, 1–8
entities and attributes, C–1
MTS populate script, B–1
MTS Transfer, 4–18
MTS Transfer Protocol, 4–18
MTS Transfer Protocol 1984, 4–18
mts_create_mts_entry.ncl, 8–5, 8–14
Multi-homed routing domain, 4–4, 5–18

N

Naming
Agents, 4–13
individuals, 5–15
MTAs, 4–9
MTA sets, 4–5
other routing domains, 4–13
your routing domain, 4–4

Naming ambiguities, 3–4
resolving, 5–16
Naming Context
for your routing domain
on OpenVMS, 9–9
on Tru64 UNIX, 8–8
Naming Context entity, 6–2
Naming scheme
choosing, 3–3
confidentiality, 3–4
extending, 3–6
planning, 3–1
recommendations, 3–3
NCL, 1–9
Nested distribution list, 5–27
Network Control Language, 1–9
Nondeliver
routing instruction, 5–20, 5–23
Non-delivery report, 2–6
Non-mnemonic O/R address
postal form, A–1
terminal form, A–5
No Parent
NCL response, 8–8, 9–9
Numeric Content Information, 5–27
Numeric O/R address, 5–3
attributes, 5–8
diagram, 5–6
Numeric Routing Instruction, 5–26
Numeric User Identifier, 5–15

O

O/R address, 5–1, 5–2
Administration Domain Name, 5–11
aliases, 5–18
and routing, 5–17
attribute, 5–5, 5–7
attribute hierarchy, 5–5
attribute string length, 5–7
complete, 5–17
conditional attributes, 5–7
country codes, 5–10
directory entry, 5–19
diagram, 7–6
O/R address (cont'd)
entries in the directory, 5–24
example, 5–16
forms, 5–7
content information, 5–27
routing instruction, 5–26
for other routing domains, 5–17
in distribution list, 5–27
information for ACME, 7–6
mandatory attributes, 5–7
mnemonic, 5–2, 5–7
numeric, 5–3, 5–7
partial, 5–7, 5–17
planning a naming scheme, 3–1
postal, 5–3
postal form, A–1
registration authority, 5–3
routing action, 5–19
routing information, 5–22
routing instruction, 5–6
rules and regulations for attributes, 5–10
table of attributes, 5–7
terminal, 5–3
terminal form, A–1
used for routing, 5–5
valid character sets, 5–7
when used for naming, 5–2
O/R address attribute
alternative spellings, 5–18
assigning, 5–15
Common Name, 5–14
Country Name, 5–10
descriptions, 5–10
Domain Defined, 5–14
for a Global Domain Identifier, 5–10
multiple values, 5–10, 5–18
Numeric User Identifier, 5–15
Organizational Unit Name, 5–13
Organization Name, 5–13
Personal Name, 5–14, 5–21
Private Domain Name, 5–12
used for routing, 5–6
O/R name, 5–1
in distribution list, 5–27

object classes, 4–3
OM_NETWORK_ERROR, 12–6
OM_SYSTEM_ERROR, 12–6
OM_WRONG_VALUE_LENGTH, 12–6
OM_WRONG_VALUE_MAKEUP, 12–6
Online Help, 1–9
OpenServer 400 and the MTA, 11–5
Open Systems Interconnection
  See OSI
ORaddress entity, 7–5
  attributes, C–2
Organization
  defining a hierarchy, 3–1
Organizational divisions, 3–1
Organizational Unit, 5–8
Organizational Unit Name, 5–13
Organization Name, 5–8, 5–13
Originator, 2–1
OSI
  layers, 1–6
  reference model, 1–6
  standards, 1–1
  transport protocol, 1–6
OSI Transport Service, 4–19, 4–22
Other routing domains
  naming, 4–13
Output Queue
  for the MTA
    on OpenVMS, 11–7
    on Tru64 UNIX, 11–6

P

P1 protocol, 2–4
P2 protocol, 2–5
Partial O/R address, 5–7, 5–17
  and routing information, 5–7
Password
  for Agent, 4–26
  for MTA, 4–20
  for other routing domains, 4–20
  for routing domain, 4–20
Peer Domain
  for Peer MTA, 4–22
Peer MTA
  Application Context, 4–18, 4–22
  authentication information, 4–19
  Dialogue mode, 4–18, 4–22
  domain name, 4–19
  entity name, 4–22
  Local Name, 4–22
  Local Password, 4–22
  Peer Domain, 4–22
  Peer Name, 4–22
  Peer Password, 4–22
  Presentation address, 4–18, 4–22
  Session address, 4–18, 4–22
  Template Name, 4–22
  Transport Service Option, 4–22
Peer MTA entity, 4–16, 7–5
  and the VAX Message Router X.400
      Gateway, 10–4
  creating, 10–4
  example, 7–13
  for boundary MTA, 4–16, 5–25
Peer Name
  for Peer MTA, 4–22
Peer Password
  for Peer MTA, 4–22
Permission information
  May Cross CCITT Boundaries attribute,
      7–10
Permission to send messages, 5–28
Personal Name, 5–8, 5–14
  and 1984 MHS Standards, 5–14
Population script, 7–5
Postal Content Information, 5–27
Postal O/R address, 5–3, A–1
  attributes used for routing, A–3
  diagram, A–3
  table, A–1
  valid character sets, A–1
  valid lengths, A–1
Postal Routing Instruction, 5–26
Presentation address
  for MTA, 4–11
  for Peer MTA, 4–22
  for Peer MTA entity, 4–18

Private Domain Name, 5–8, 5–12
in a GDI, 4–4
rules and regulations, 5–12
Private Management Domain, 5–4
PRMD, 5–4
See Private Domain Name
Probe, 2–7

R

Recipient, 2–1
Recipient Redirection
routing instruction, 5–20
Recommendations
X.400, 1–2
Recovery, 5–29
Redirect
routing instruction
MD, 5–23
recipient, 5–20
Redirection, 5–20, 5–23
Registered Agent, 4–12
Registration authorities, 5–3
Remote management, 1–7
Replicating routing information, 7–2
Replication, 6–1
guidelines for, 6–4
Report, 2–6
delivery, 2–6
non-delivery, 2–6
structure, 2–6
Resolving naming ambiguities, 5–16
RFC 1006, 1–6
RFC 1006 NSAPs, 8–5, 8–14, 9–6, 9–17
Routing, 1–3
based on area, 4–7
using the directory, 5–5
Routing domain
definition, 4–1
diagram, 4–2
Different CCITT Domain, 4–22
dividing into areas, 4–7
Global Domain Identifier, 4–20
intermediate, 4–15
name, 4–3, 4–20
Routing domain (cont'd)
other, 4–13
password, 4–3, 4–20
summary, 4–3
Routing domain entry
diagram, 4–16, 7–11
in the directory, 4–15
position in HP Enterprise Directory Service, 4–3
Routing domain layout
diagram, 4–5
Routing domain name, 4–16
in the directory, 4–4
Routing information, 5–7
and the directory, 5–6
diagram, 5–21, 5–23
ensuring availability, 6–3
for complete O/R addresses, 5–19
for different forms of O/R address, 5–26
for O/R addresses, 5–22
for other routing domains, 5–25
for your routing domain, 5–21
held in the directory, 7–1, 7–5
held locally at an MTA, 7–1, 7–4
important concepts, 7–1
remote access, 8–19, 9–23
setting up, 10–2
stored in the directory, 5–6
Routing instruction
Definitive O/R Address, 5–20, 7–11
Deliver, 5–19, 5–23, 5–25, 7–10
forms of, 5–26
for routing domains, 5–25
MD Redirect, 5–23, 7–8
Nondeliver, 5–20, 5–23
Personal Name, 5–21
Recipient Redirect, 5–20
Transfer Through Domain, 5–26, 7–12
Transfer To Domain, 5–20, 5–23, 5–25, 7–8, 7–12

S

Server MTA
  for Area, 4–20
Session address
  for Peer MTA, 4–22
  for Peer MTA entity, 4–18
Setting up Agents, 11–1
Setting up an MTA
  running MTA Setup
    on OpenVMS, 9–5, 9–15
    on Tru64 UNIX, 8–4, 8–13
Setting up Shared File 1984 Agents, 11–5
Setting up Shared File 1992 Agents, 11–4
Setting up XAPI Agents, 11–1
Setting up your routing domain
  on OpenVMS, 9–1
  on Tru64 UNIX, 8–1
Setup
  for the first MTA
    on OpenVMS, 9–8
    on Tru64 UNIX, 8–7
  privileges
    on OpenVMS, 9–3
    on Tru64 UNIX, 8–2
  when upgrading
    on OpenVMS, 9–4
    on Tru64 UNIX, 8–3
Shadow copies, 6–2
Shadow DSA, 7–2
Shared File 1984 Agents
  setup, 11–5
Shared File 1984 interface
  on OpenVMS, 11–7
  on Tru64 UNIX, 11–6
  queues, 11–6
Shared File 1992 Agents
  setup, 11–4
Shared File 1992 interface
  on OpenVMS, 11–7
  on Tru64 UNIX, 11–6
  queues, 11–6
Shared File interface, 4–12
  for Gateways, 4–12
Standards
  ISO/IEC 10021, 1–2
  ISO 9594, 1–3
  X.400, 1–2
  X.500, 1–3
Starting the MTA
  on OpenVMS, 9–7, 9–18
  on Tru64 UNIX, 8–6, 8–15
start_mta.ncl, 8–5, 8–14
Store-and-forward, 2–2
Structure of an MTA name
  diagram, 4–9
Submission, 2–3
Subordinate Reference entity, 6–3
Surname, 5–8

T

TCP/IP, 1–6
  and MTA setup, 8–5, 8–14, 9–6, 9–17
TCP/IP Transport Service, 4–19, 4–22
Template Name
  for Peer MTA, 4–22
Terminal Content Information, 5–27
Terminal O/R address, 5–3, A–5
  attributes used for routing, A–7
  diagram, A–7
  table, A–5
  valid character sets, A–5
  valid lengths, A–5
Terminal Routing Instruction, 5–26
Transfer Through Domain
  routing instruction, 5–26, 7–12
Transfer To Domain
  routing instruction, 5–20, 5–23, 5–25, 7–8, 7–12
Transport protocol, 1–6
Transport Service
  OSI, 4–19, 4–22
  TCP/IP, 4–19, 4–22
Transport Service Option
  for Peer MTA, 4–22

Transport Template, 4–19
Type
  for Agent, 4–26

U

Unregistered Agent, 4–13
Upgrading
  and MTA setup
    on OpenVMS, 9–4
    on Tru64 UNIX, 8–3
  and starting MTA
    on OpenVMS, 9–6
    on Tru64 UNIX, 8–4
User Agent, 1–1, 4–13
  diagram, 2–2
  role of, 2–3
Using the directory, 6–1
Using the MTS entity, 7–2

V

Verification procedure
  on OpenVMS, 9–20
  on Tru64 UNIX, 8–16

W

Warning text, 4–20, 5–29
Worksheet
  for Agent entity, 4–26
  for Peer MTA entity, 4–22
  for routing domain layout, 4–20

X

X.25 Template, 4–19, 8–7, 8–16, 9–7, 9–19
X.400
  definition used in the MTA documentation, 1–2
  management domain, 5–4
  Recommendations, 1–2
X.400 management domain, 4–4
  permission to send mail, 5–28
X.500
  See Directory
X.500 distinguished name
  for the routing domain, 4–4
XAPI Agents
  setup, 11–1
XAPI interface, 4–12
  for Agents, 4–12