Internet Exchange
for cc:Mail

Gateway Administrator's Manual

Version 2.0
May 1996

International Messaging Associates Limited
New Trade Plaza
Block A
6/F, Suite 7
6 On Ping Street
Shatin, N.T.
HONG KONG

Tel: +852 2649-0135
Fax: +852 2648-5913
Email: *info@ima.com*
WWW: *http://www.ima.com/*

ISBN: 962-8137-01-8

Internet Exchange version 2.0 is dedicated to the memory of Brent Stilley.

TABLE OF CONTENTS

PREFACE

OVERVIEW

Internet Exchange is an electronic mail gateway that connects Lotus cc:Mail Local Area Network email environments to either the Internet, and/or to private TCP/IP based Local Area Networks. **Internet Exchange** is the most advanced and complete gateway technology linking cc:Mail users with the Internet.

This manual describes two versions of **Internet Exchange**: the *Enterprise Edition* (unlimited usage), and the *Workgroup Edition* (100-user). With the exception of the number of users restriction, default address mapping options, and default send/receive permissions in the *Workgroup Edition*, the two versions are identical. The few areas where there are differences are noted in this manual.

Throughout this manual, the term "Internet" is considered to be interchangeable with any TCP/IP based network or collection of networks.

This manual describes how to configure, use, and administer **Internet Exchange**. It contains the following sections:

- **Part 1**, "*Introduction*" introduces the technologies in **Internet Exchange**. The overall architecture is presented, and introductions to electronic mail addressing, the TCP/IP protocol stack, SMTP, MIME, Apple Macintosh file structure and attachment naming conventions.

- **Part 2**, "*Installation*", describes how to install **Internet Exchange**.

- **Part 3**, "*Operation*," describes how to set up and use **Internet Exchange** including configuration, initialization, rules based addressing, peer domain configuration and gateway monitoring.

- **Appendices** - The appendices present more detailed information on topics related to **Internet Exchange** including MIME, TCP/IP and the Domain Name System (DNS).

REQUIREMENTS

This section describes the hardware and software requirements necessary to run **Internet Exchange**.

Hardware

The minimum hardware requirements for **Internet Exchange** are:

- **Processor:** 80386 or higher
- **Memory:** 8MB (Windows 3.1), 16MB (Windows 95), 20MB (Windows NT Workstation), 24MB (Windows NT Server)
- **Hard Disk:** 200 MB or more
- **Monitor:** Any Windows VGA resolution or higher monitor
- **Mouse:** Any Windows-supported mouse
- **Network Interface Adapter:** Any network interface board that is hardware compatible with your TCP/IP network, with support for at least one of the following drivers: ODI, NDIS, or packet drivers. This network interface board will communicate with other hosts on your TCP/IP network. In addition, if the connection to your cc:Mail Post Office is over a LAN with a different cabling system than your TCP/IP network, you will need a separate Network Interface Adapter for that network as well.

Software

The software requirements for **Internet Exchange** are as follows:

- **Operating System:** One of the following:
    - Microsoft Windows NT Server 3.5 or later
    - Microsoft Windows 95 or later
    - Microsoft Windows for Workgroups 3.11 or later
    - Microsoft Windows 3.1
- **TCP/IP:** TCP/IP protocol stack compliant with WINSOCK Version 1.1. Microsoft Windows NT Server 3.5 and Windows 95 both come with built-in stacks that meet this requirement. A stack is also available from Microsoft for the Windows for Workgroups environment.
- **VIM:** Lotus VIM (version 2.07 or later) for cc:Mail. The VIM (Vendor Independent Messaging) library is used to read from and write to the cc:Mail Post Office.

INSTALLATION

Internet Exchange is a network application that operates in a Microsoft Windows environment. It requires the ability to communicate with the TCP/IP network via a WINSOCK Version 1.1 compliant interface. **Internet Exchange** also requires a Lotus cc:Mail VIM Version 2.07 or greater interface to communicate with the cc:Mail Post Office.

To install **Internet Exchange**, the administrator must perform the following tasks:

1. Locate and install all hardware and software shown in the table above.
2. Install and configure the software necessary to communicate with your LAN Network Operating System. Examples of common Network Operating Systems include NetWare ®, LANtastic ®, LAN Manager ®, and others.
3. Install and configure the software necessary to communicate with your TCP/IP network.
4. Make sure the proper VIM cc:Mail libraries have been installed and accessible to the gateway machine. See Chapter 3.
5. Install **Internet Exchange**. See Chapter 3.

CONVENTIONS USED IN THIS MANUAL

This manual uses the following conventions:

Example	**Description**
`text in Courier font`	Used to show text in commands, listings, and files. It also shows text to type as it appears.
`Italics`	Used to show variables that represent actual names or addresses that you enter.

To provide addressing examples, we have taken the case of an organization called Jade Networks, which uses the fully qualified domain name of jade.net, and operates an Internet Exchange gateway with the fully qualified domain name of iegate.jade.net. Examples involving remote sites use a company called XYZ Corp, with a FQDN of xyz.org.

PART 1 INTRODUCTION

CHAPTER 1
INTERNET EXCHANGE FUNCTIONAL OVERVIEW

INTRODUCTION

Internet Exchange acts as a bridge between a cc:Mail Post Office and the Internet (or any TCP/IP network). The relationship between the **Internet Exchange** gateway, the cc:Mail environment and the Internet (or your local TCP/IP network) is shown in Figure 1.1.

Internet Exchange takes messages from the cc:Mail environment and converts the message as well as user addresses into formats understood in the Internet community. In the other direction, **Internet Exchange** takes messages and addresses in the format understood on the Internet and converts them into messages and addresses that can be dealt with in the cc:Mail environment. This conversion process is transparent to end users on both sides of the gateway. Because of the translation done at the gateway, cc:Mail users appear as Internet users on the Internet side of the gateway, and Internet users appear as cc:Mail users in the cc:Mail environment.

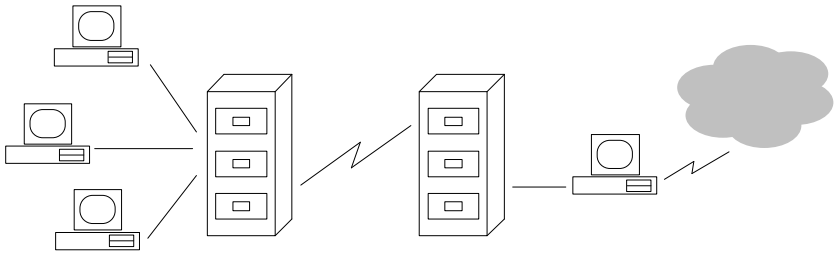


Figure 1.1 ***Internet Exchange*** *Bridging cc:Mail and the Internet*

INTERNET EXCHANGE ARCHITECTURE

Internet Exchange is a multiprocess, multitasking gateway. Its various components operate asynchronously with respect to each other. This allows the gateway to handle many tasks at the same time. Since the communications channels are typically much slower than the host processor, the gateway has the ability to service several channels at the same time. In addition, message and address conversion can be done while other messages are being transferred. This multiprocessing capability has the effect of maximizing overall gateway throughput.

Internet Exchange is divided into the following functional components:

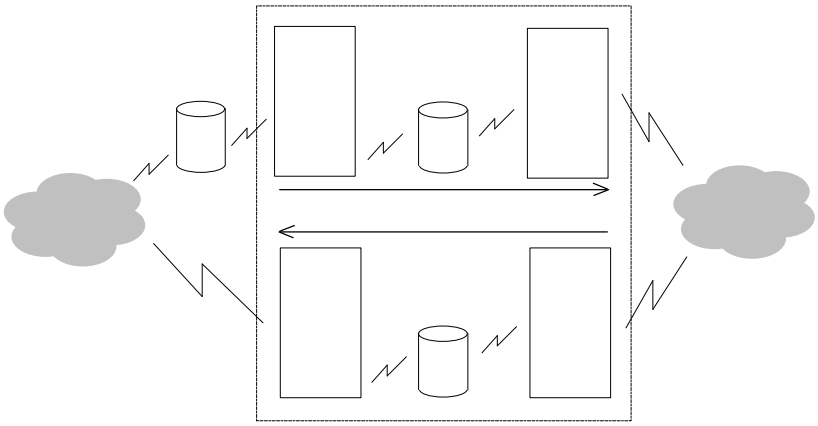


Figure 1.2 ***Internet Exchange*** *Functional Block Diagram*

cc:Mail Gateway Post Office

When cc:Mail users send mail to users on the Internet, they do so by sending the mail to the cc:Mail remote post office name that is assigned to the gateway. The cc:Mail post office that acts as the mail forwarder will contain an entry for the gateway post office in its directory. Mail that arrives for the gateway post office will be temporarily stored in the gateway post office mailbox.

In the cc:Mail environment, **Internet Exchange** acts as the gateway post office. Like all other cc:Mail post offices, it is given a unique name, typically *Internet* (although any unique name will work). **Internet Exchange** regularly polls the gateway mailbox on the forwarding post office. This ensures the regular pickup of mail leaving the cc:Mail domain for the Internet.

Internet Exchange is also responsible for regularly checking for inbound messages. It delivers any messages found to the forwarding post office for either final delivery to a user mailbox or for further routing within the cc:Mail domain.

Interaction with the cc:Mail post office

Internet Exchange communicates with cc:Mail using the VIM (Vendor Independent Messaging) protocol. There are two separate programs that talk to the cc:Mail post office, one of which imports messages from the Internet, and one that exports messages from cc:Mail bound for the Internet.

Interaction with the Internet

Internet Exchange communicates with mail hosts on the Internet using the **S**imple **M**ail **T**ransfer **P**rotocol (SMTP). This protocol is used for the submission as well as the reception of mail messages. **Internet Exchange** implements SMTP as two separate modules. A client program sends messages from the gateway to the Internet, and a server program receives messages from the Internet bound for cc:Mail.

Outgoing Mail

Exporting cc:Mail messages to the Internet involves two queue processors: *CCOUT* and *SMTPC.*

CCOUT

CCOUT obtains messages from cc:Mail by polling the gateway post office. The gateway administrator determines the polling interval, which can be set for as often as once every minute.

The module that performs the polling of the cc:Mail gateway post office and transfers messages into the gateway is *CCOUT*. *CCOUT* logs into the post office using the cc:Mail VIM interface. If messages are present, it will move each message, one at a time out of the gateway post office and into an internal gateway queue (*SMTP OUT*). In the process of moving a message into the *SMTP OUT* queue, *CCOUT* will perform any address and message format translations necessary. *CCOUT* is also responsible for creation of the initial SMTP envelope.

SMTPC

The SMTP client program (*SMTPC)* is responsible for delivery of messages on the Internet. It does so by regularly checking for messages queued in the *SMTP OUT* queue. When messages are found, it establishes one or more connections with external SMTP servers and transfers the message to the appropriate Internet mail host.

The **Internet Exchange** SMTP client program is capable of routing Internet mail messages based on several criteria. The routing options are:

- Domain Name System (DNS) host name lookup
- Host Table lookup of destination host
- DNS followed by Host Table lookup
- Host Table followed by DNS lookup
- Delivery to default mail relay host(s)

Mail routing via the DNS is the preferred method of mail routing in the Internet. The DNS is an Internet network service that provides for the storage and retrieval of information associated with domain names. In the context of Internet mail, the records that are of interest are mail exchanger (MX) records and address (A) records.

MX-records are used to store mail forwarder information for hosts registered in the Internet. An MX-record will contain the name of the host or domain, and a list of one or more mail forwarding hosts and the preference values associated with these hosts. The preference values are used by *SMTPC* to determine the order to attempt delivery in the case where more than one mail forwarder has been identified. MX-records are essential for the proper routing of mail, especially in situations where the destination host is not physically connected to the Internet and has to rely upon a mail forwarder for proper mail delivery. As an example, many organizations rely upon the UUCP communications package which comes with the UNIX operating system to physically exchange mail. These sites, through the use of MX-records can appear to be connected to the Internet, even though mail is the only Internet service they use.

A-records are used to store Internet address information for hosts. When configured to use the DNS, **Internet Exchange** first attempts to obtain an MX-record for the

destination host. If an MX-record is found, the list of mail forwarding hosts is used when the SMTP connection is attempted. If no MX-record can be found, **Internet Exchange** searches for an A-record. If an A-record is found, then this address is used when the SMTP connection is established.

If the SMTP client is configured to use host table lookup, the internal host table, usually a text file, is used to determine the Internet IP address of the recipient host. The exact format and path name of the host table depends upon the TCP implementation. The location of the host table is specified when **Internet Exchange** is installed (see Chapter 3). This is the equivalent of doing an A-record lookup using the DNS. However most internal host tables are nowhere near as complete a database as that which the DNS can provide.

When configured to use a default mail relay host, all messages will be sent to a primary mail forwarder for further routing. If this mail forwarder cannot be contacted for any reason, and a secondary mail relay host is defined, the gateway will use the secondary mail relay. In this case, It will occasionally check to see if and when it is possible to switch back to use the primary relay host. Use of this option will improve gateway throughput, as mail forwarding hosts are usually on the same network as the gateway. Response time and throughput are typically fast, resulting in little to no backlog of messages at the gateway. The use of this option, however, places the burden of routing and retries of delayed messages on the mail forwarding machine(s), which will add to their existing workload.

Internet Exchange can be configured to use a combination of the above strategies to deliver mail. When not using a mail relay, it is recommended to use a strategy where the DNS is consulted first, and then a local host table in the event of a failure to resolve a name with the DNS. The opposite configuration can also be used if needed. In any event, if the name cannot be resolved using either of the above methods, **Internet Exchange** will fall back to using the mail relay host(s) as the next hop (assuming there is at least one configured), in the hope that resolution can be better handled at that site.

Incoming Mail

Importing Internet messages to cc:Mail involves two queue processors: *SMTPD* and *CCIN*.

SMTPD

The **Internet Exchange** module that receives messages from the Internet for cc:Mail users is the SMTP daemon *SMTPD*. Unlike the other **Internet Exchange** modules, *SMTPD* continuously runs on the gateway machine. This is necessary because there is no way for the gateway to predict the timing or frequency of inbound messages.

SMTPD is a background Windows process that listens for incoming SMTP connections on TCP port 25. When a connection request is detected, it creates a new sub-process that manages the new connection. *SMTPD* is capable of maintaining many simultaneous SMTP connections, the maximum number being a configurable parameter that can be set based upon the performance of the underlying TCP/IP stack. The maximum permitted is 40 concurrent sessions, due to a Windows file handle limit. This ability to handle concurrent SMTP sessions reduces delay in message delivery as remote mail forwarders do not have to wait for an existing SMTP session to complete.

When a message is received by *SMTPD*, it is placed in the gateway queue *SMTP IN*. *SMTPD* does not perform any message translation. It simply creates the queue entry and goes back to wait for additional connection requests.

CCIN

Mail messages are delivered to the cc:Mail environment by the **Internet Exchange** module *CCIN*. *CCIN* is run by *SYSMAN* at regular intervals, and is responsible for moving messages between the *SMTP IN* queue and the cc:Mail post office. The frequency at which *CCIN* is run is a configuration option set by the gateway administrator.

When *CCIN* is run, it first checks the *SMTP IN* queue for any messages. For each message that it finds, it performs address and possibly message content translation in preparation for submission to cc:Mail. Addresses are converted to a form that can be used in the cc:Mail environment. For messages that are multipart or are encoded non-textual data, message conversion is performed by *CCIN* prior to delivery to cc:Mail.

Once the message is properly converted, *CCIN* logs into the gateway cc:Mail post office, and delivers the message. The message is either directly delivered to a local cc:Mail user, or further routed within the cc:Mail domain.

Internet Exchange System Manager

The **Internet Exchange** System Manager (*SYSMAN*) is the front end that allows configuration of the gateway, as well as the regular scheduling of gateway activity. The System Manager keeps track of the times and frequency at which the various queue managers (*CCIN, CCOUT, SMTPC*) are run, and is responsible for the launching of these managers.

In addition to being the glue that ties the different **Internet Exchange** modules together, the System Manager is used to configure and administer the gateway. All configuration options, including those associated with the cc:Mail post office, message routing strategy, Internet host name, scheduling, MIME file mapping, and user address mapping are handled by the System Manager.

The System Manager also provides views of each of the three gateway queues: *SMTP IN*, *SMTP OUT*, and the outbound cc:Mail post office mailbox. The administrator can obtain detailed information for any message in any queue by selecting the appropriate queue, then double-clicking on the message item. Once selected, individual messages can be forwarded, bounced, delivered, or removed from the queue by simply selecting the appropriate button from the main screen.

ATTACHMENT HANDLING

One of the main differences between the Internet mail environment and cc:Mail is in how attachments and non-text messages are handled. Within the cc:Mail environment, users with non-text attachments exchange messages in their native format. It is not necessary to perform any translation on the messages prior to submission to the message transport.

On the Internet, this is not the case. RFC-821 (Simple Mail Transfer Protocol) which defines SMTP, places certain restrictions on messages. These restrictions include certain line length limits as well as the restriction that all data be 7-bit ASCII characters. Because of these restrictions, it is not possible to transfer arbitrary objects using SMTP unless these objects are encoded prior to being submitted to SMTP.

The Internet standard for specifying how to encode non-textual and multipart messages is called MIME (Multipurpose Internet Mail Extensions) and is defined in RFC-1521 (Mechanisms for Specifying and Describing the Format of Internet Message Bodies).

This Internet standard not only defines certain encoding and decoding methods, but also the format in which attachments are to be labeled and identified within the message.

When the MIME specification initially came out, it addressed many of the non-textual file attachment problems present within Internet email. One area that was not adequately addressed was the handling of Apple Macintosh file types (see Chapter 3 for additional information on the differences between these files and DOS or UNIX file types). Shortly after the MIME specification was produced, RFC-1740 (MIME Encapsulation of Macintosh files - MacMIME) was developed to address this shortcoming. By using both the MIME and MacMIME specifications, it is now possible to exchange information between Macintosh and non-Macintosh environments by email.

Prior to MIME, there were no official Internet standards in the areas of encoding standards and multipart message representation. There were some RFCs dealing with multipart message representation, but none of these ever made it to official standard status within the IETF. As a result, several vendors decided to implement one or more of the experimental RFCs covering this area, or to invent their own mechanism for specifying multipart messages.

One of the pre-MIME products that came to market was the Link to SMTP for cc:Mail by Lotus. This product uses uuencode and uudecode for the encoding and decoding of individual message attachments.

Internet Exchange is capable of identifying and decoding incoming messages that conform to the MIME and MacMIME standards as well as the convention used in the Lotus gateway. For multipart messages that **Internet Exchange** creates, both MIME as well as non-MIME compliant messages can be generated. Messages that conform to the MIME/MacMIME standards are produced by default, however it is also possible to define peer domains where different encoding methods can be used for different destination machines or domains. This capability allows **Internet Exchange** to communicate effectively with Internet sites running MIME compliant software as well as with older sites that are not MIME conformant.

INTERNAL DATABASES

Internet Exchange uses several internal Btrieve databases for improved performance. These databases include the following:

Name	Database File	Backup Text File
Alias Database	smtpadr.btr	smtp.adr
Domain Database	smtppod.btr	smtp.pod
Directory Database	rulebadr.btr	- *none* -
Magic Database	magic.btr	[magic] in ima.ini
DNS Cache	dns.btr	- *none* -
Message Database	mesg.btr	- *none* -
Peer Database	peer.btr	- *none* -

Alias Database

The Alias Database stores specific cc:Mail name to Internet name mappings as well as the corresponding send and receive permissions associated with specific cc:Mail users. It is used primarily by *CCIN* and *CCOUT* for cases where either the application of Rules Based Addressing or the default send/receive permissions does not achieve the desired result. The Alias Database takes priority during address translation over the Directory Database as well as the Default Encoding method selected for the gateway.

Domain Database

The Domain Database provides for the mapping between downstream cc:Mail post offices and Internet domain names. This allows downstream post offices to have unique Internet domain names associated with them.

Directory Database

The **Internet Exchange** *Rules Compiler* produces the Directory Database by compiling the Rules Based Addressing rules. When Rules Based Addressing is selected by the administrator, the selected rules will be applied to the contents of the cc:Mail directory. This results in the creation of the **Internet Exchange** Directory Database, a shadow version of the cc:Mail Directory available from the local Post Office. This database is then consulted by *CCIN* and *CCOUT* when performing routine address translation tasks, eliminating most references to the cc:Mail Directory. This produces higher throughput for the gateway since fewer relatively slow cc:Mail Directory lookups need to be performed.

Magic Database

The Magic Database is used to store the default encoding rules for various types of file attachments. In previous versions of **Internet Exchange** this information was stored in the *IMA.INI* file, but has been moved to its own database to increase performance.

DNS Cache Database

The DNS Cache Database is used by *SMTPC* to store DNS query results. Each time **Internet Exchange** needs to perform a host name or MX record lookup, this will normally result in a DNS query over the network. When DNS Caching is enabled, the results from these queries are locally stored. The next time a query for the same host is made the **Internet Exchange** DNS resolver code will first check the local cache for the information, eliminating the time consuming network DNS lookup. This both lowers network congestion as well as significantly increasing gateway performance, especially for sites where the DNS server is not connected to a high speed network.

Message Database

The Message Database is used by **Internet Exchange** to store messages as they travel through the gateway. This database manages the inbound as well as outbound SMTP queues. Three basic units of information are stored for each message as it passes through: the message status, the envelope (list of recipients), and the actual message. The latter is actually stored in a separate flat file, one file per message, with the Message Database maintaining pointers to the message files.

Peer Database

The Peer Database is used to store information about remote hosts or domains that either require different encoding rules or send/receive permissions than the defaults specified for the gateway. This is typically used to identify remote companies or organizations that are still running pre-MIME software and unable to effectively deal with MIME attachments. It can also be used to restrict email access in one or both directions remote to/from specific remote sites.

An **Internet Exchange** *peer* is defined to be a remote host or domain name. In the case of a domain, the scope of a particular *peer* definition includes the peer domain name as well as all names and subdomains of that peer.

Peer definitions are processed from the most specific to the most general. A peer definition for a subdomain of a previously defined domain will take precedence over the more general definition.

As an example, let's say you want to configure the mail encoding for a company called XYZ Corp with a domain name of *xyz.org*. Most of the users in this company are still running pre-MIME software and prefer to receive non-MIME encoded messages. One group however, the engineering group, is running their group with MIME compliant software. If the engineering group creates their own subdomain, say *engr.xyz.org*, then you can setup two peer definitions for XYZ Corp.

The first peer definition will define how you want to encode mail for everyone except the Engineering group. You do this by defining a the peer *xyz.org*, and specifying non-MIME encoding for this domain. You then setup a second peer for *engr.xyz.org*, where you specify MIME compatible encodings. This has the effect of sending MIME messages to the Engineering Department within XYZ Corp while at the same time sending non-MIME messages to everyone else within the organization.

Hierarchical peer definitions can be added without limit, producing any combination desired. Peer definitions do not have to be associated with domains as in the above example either. The remote peer can be a machine or gateway name, providing configurability all the way down to the individual host level.

ADDRESS TRANSLATION

In order for users to exchange messages between their cc:Mail environment and the Internet it is necessary that they be able to uniquely identify both themselves as well as their intended recipients on both sides of the gateway. Each user on either side of the gateway has an electronic mail address associated with each mailbox. Unfortunately, the formats of these mail addresses are not the same for cc:Mail users and Internet mail addresses. Due to the differences between the two addressing formats, it is the job of the gateway to provide address mapping capabilities.

Internet Exchange offers several different options when it comes to translating between Internet and cc:Mail email addresses. In cc:Mail many if not most users have spaces between their first and last names. This aids in readability and allows users to use their real names, without any artificial limitations on length or case. On the Internet, user addresses are more restricted. Spaces are not allowed, and addresses are case sensitive.

Internet Exchange has four mechanisms to map between user names in cc:Mail, and the corresponding Internet addresses by which these users will be known to other Internet users:

- Default Mappings
- User Aliases

- Post Office Subdomains
- Rules Based Addressing

The order in which the addressing methods are applied is: user aliasing, rules based addressing , domain mapping , and finally default mapping .

If an outgoing address already contains a separator character before the default mapping is applied, this separator will be doubled. When a reply to such an address is received, the extra separator will be discarded. e.g. if the separator is underscore (_), then the following user:

John_Doe at Remote Sales_PO

will be mapped to:

John__Doe_at_Remote_Sales__PO

Default Mappings

The default method of creating an Internet address from a cc:Mail username is to convert all spaces to underscores. e.g. the user:

John Doe at Main PO

becomes:

John_Doe@iegate.jade.net

Users of cc:Mail on remote Post Offices (the Post Office named *Sales* in the following example) will appear similar to:

Jane_Doe_at_Sales@iegate.jade.net

The remote Post Office name is included so that cc:Mail can reroute replies to this message. Although these address forms are perfectly adequate, often users wish to be known by different Internet addresses. The administrator can achieve this by creating a user alias.

User Aliases

By entering a mapping in the Configure Users screen, an alias can be created for a cc:Mail user. This is a simple textual substitution that happens for both outgoing and incoming messages. For example, an alias for John Doe can be created, mapping him to *johnd*. Any messages from this user will appear to come from johnd@iegate.jade.net. Similarly, replies to such messages will be converted back to the original cc:Mail user name.

These user aliases are most easily created by the Configure Users screen. An alternative method is to edit the file *SMTP.ADR*, which contains these mappings. The above example appears in this file as follows:

John Doe<=>johnd

An equivalent form is

John Doe<=>johnd@iegate.jade.net

Although effectively the same, the first format is preferable, as it will not need changing if the local hostname and/or domain is ever changed. When there is no hostname and domain in an alias, the local values are appended automatically.

Using the manual method can be useful if a list of aliases can be created automatically by a program. Then, the resulting text output can easily be converted into the format needed for *SMTP.ADR*.

Post Office Subdomains

If there are a number of cc:Mail Post Offices in use, then the default method of creating an Internet address produces ungainly results. e.g. John Doe at Sales becomes:

John_Doe_at_Sales@iegate. jade.net

Another mechanism is available that involves mapping remote cc:Mail Post Offices to Internet subdomains. This is setup by the *Configure Domains* screen. e.g. by mapping Sales above to sales.iegate.firm.com, the above address appears as:

John_Doe@sales.iegate. jade.net

These domain mappings are most easily created through the Configure Domains screen. An alternative method is to edit the file *SMTP.POD*, which contains these mappings. The above example appears in this file as follows:

Sales = sales.iegate.jade.net

Using the manual method can be useful if a list of mappings can be created automatically by a program. Then, the resulting text output can easily be converted into the format needed for *SMTP.POD*.

Addressing Options

There are also several options that affect how addresses are formed. They are accessed through the Configure Options screen as follows:

Addressing Separator

This allows the administrator to choose between the underscore (_) and the dot (.) as the separator used in default mapping. This option should be set initially and not changed, otherwise incoming messages using default addressing will be bounced.

Include cc:Mail Names In Addresses

Setting this option appends the cc:Mail user name in parentheses to any Internet address generated.

Use Hostname In Addresses

This option determines the format of the local hostname used for outgoing Internet addresses. e.g. if set, addresses will look like:

User@iegate.jade.net

and if not set:

User@jade.net

Organizations who wish to hide the internal structure of their network from the outside world often prefer the second format.

Use Remote PO Names

When set, the Post Office name of a user from a remote cc:Mail Post Office is included in the default address mappings. e.g.:

John_Doe_at_Sales@iegate.jade.net

If reset, the above address appears as follows:

John_Doe@iegate.jade.net

This format is much cleaner and is an alternative to using Post Office subdomains. However, if this format is used, the main cc:Mail Post Office must contain entries for all cc:Mail users from other Post Offices. Otherwise, incoming messages will not be able to be routed internally to the correct Post Office. An easy way to ensure this occurs is by running Lotus Automatic Directory Exchange (ADE).

Summary of Default Addressing Formats

The following table lists the options to set depending upon which form of default addressing is preferred. The user here is John Doe at Sales, a remote Post Office:

NAMES	***ADDRESS SEPARATOR***	***USE HOST NAME***	***USE REMOTE PO***
John_Doe_at_Sales@iegate.jade.net	underscore	yes	yes
John_Doe_at_Sales@jade.net	underscore	no	yes
John_Doe@iegate.jade.net	underscore	yes	no
John_Doe@jade.net	underscore	no	no
John.Doe.at.Sales@iegate.jade.net	dot	yes	yes
John.Doe.at.Sales@jade.net	dot	no	yes
John.Doe@iegate.jade.net	dot	yes	no
John.Doe@jade.net	dot	no	no

NOTE: if not using remote PO names, it is best to use ADE .

Rules Based Addressing

For large sites that prefer to map cc:Mail user names to Internet addresses that do not match the rules used by default address translation, it can be inconvenient and time consuming to maintain large alias translation tables. To address this problem, **Internet Exchange** also makes use of administrator specified address translation rules.

The **Internet Exchange** *Rules Editor* provides the gateway administrator with the ability to specify an ordered set of rules that the gateway can use when trying to map Internet email addresses to their cc:Mail user name counterparts. The gateway applies each rule in order, allowing a site to effectively support several different naming conventions without forcing each name mapping to be manually entered into the system.

As an example of how rules based address translation can be used, let's once again look at Jade Networks. The company policy for Jade Networks is that user's Internet email addresses be of the form first initial followed by the first 7 characters of the last or family name. In addition, some people in the company also want their Internet email address using simply their family name, regardless of length.

Using the *Rules Editor*, the gateway administrator defines two rules. The first specifies the primary company convention of first initial followed by no more than the first 7 characters of the family name. A second rule is then defined to map to the family name, regardless of length. After the rules are entered, the *Rules Compiler* is run, which reads the post office directory, applying the rules to produce an **Internet Exchange** shadow directory mapping database.

Once the shadow directory mapping database is created, it will be consulted prior to any attempt to perform default address translation. There are several benefits for this type of configuration:

- Since the rules are defined by the administrator rather than the software, many more combinations are available, decreasing the need for large alias databases.

- Users can accept email addressed to several different Internet email addresses, as long as they conform to the conventions defined by the organization.

- The use of the shadow directory mapping database at runtime significantly reduces the need for the gateway to access the cc:Mail directory. This results in improved performance in the gateway and less overhead for the post office file system.

Additional information on rules based addressing and its configuration can be found in Chapter 8.

CHAPTER 2
ELECTRONIC MAIL ADDRESSING

INTERNET MAIL ADDRESSING AND ROUTING

RFC-822 Message Format

An RFC-822 message is the standard unit of electronic mail on the Internet, and with mail gateways that exchange mail with Internet-connected systems. When the basic Internet mail standards were published in 1982, RFC-821 (Simple Mail Transport Protocol) defined the method of exchanging messages among Internet hosts. RFC-822 (Standard for the Format of ARPA Internet Text Messages) defined the overall format of the messages themselves, as well as the syntax and semantics of mail addressing.

Each RFC-822 message is made up of simple ASCII text consisting of two parts: a header and a body. (This entire message is enclosed in a so-called "envelope," which the SMTP client and server use to route and deliver the message.) The message header consists of lines of keywords, usually individually called "headers," followed by a colon (':') and a text or numeric string value. The full list of keywords can be found in RFC-822, and is further extended by the MIME specification (see Appendix A). The headers are separated from the message body by a blank line; the body is an ASCII text of arbitrary length.

Of the required RFC-822 headers, the most important for mail addressing and routing are the From:, To:, Cc: and Bcc:, Sender:, and Reply-To: headers (and variations on these indicating a re-sent [forwarded] message). Each of these is expected to contain one or more valid mailbox names (e-mail addresses) in the form specified by RFC-822.

From: / Resent-From:	Identity of the person or process that caused this message to be sent. Should be a single mail address. "Resent-From:" indicates a forwarded message.

Sender: / Resent-Sender:	Identity of the person or process actually sending the message. (This could be a secretary or a software agent.) Optional if the contents of the field would be identical to the "From:" line.
Reply-To: / Resent-Reply-To:	The address to which replies to this message should be sent. Redundant if contents would be the same as the "From:" header.
To: / Resent-To:	Identity of the primary recipients of the message.
Cc: / Resent-Cc:	Identity of the secondary (informational) recipients of the message.
Bcc:	Identity of additional ("blind") recipients of the message. The contents of the field are not included in the message header but are processed separately by the sender's mail agent.

Some additional RFC-822 header fields of interest include "Message-ID" (which uniquely identifies the message, usually by combining a time-stamp and serial number with the sender's system's domain name); "Received" (a time-stamped "postmark" added by each system that the message passes through, for tracing and debugging purposes); "Subject" (the topic of the message); and "Date" (a time-stamp for the message).

In addition, users may define additional header fields for private use and place them in the header. These headers begin with the letters "X-", e.g., "X-Full-Name", and are guaranteed not to conflict with RFC-822 headers defined in the future.

Since RFC-822 was published, there have been some extensions to the set of defined header fields. The most important of these are the MIME headers, which are discussed in Appendix A.

RFC-822 Addresses

An RFC-822 mailbox address is made up of two parts — on the left is the "local-part", and on the right is the "domain-part". They are joined in the middle by the "@" symbol, pronounced "at". (Example: roger@wilco.org.) The domain-part specifies how the message should be routed on the Internet, and the local-part specifies how it should be delivered when it arrives at its destination. SMTP clients and servers and other mail transport agents and gateways normally only deal with the domain-part. Other programs called *local delivery agents* use the local-part to determine how to deliver the message to a particular mailbox, program, or mail filter. (Sometimes further transport of the

message to another mail system is “hidden” in the local-part. This is discussed below.)

The content of the domain-part determines how the message will be routed on the Internet. RFC-822 discusses a number of domain ideas, but predates the adoption of the Domain Name System (DNS). It thus should not be used as an authority for domain syntax and semantics, in which it has been superseded by STD 13, (RFC-1034 and RFC-1035).

RFC-822 permits comments and explanatory material in the contents of originator and recipient header fields. These are often used to note the actual names of users corresponding to mailbox addresses, the names of mailing lists or mail aliases, or the names or versions of programs that send or receive mail. Strings inside parentheses are always comments; arbitrary strings not containing syntactically significant characters may also appear as comments on a line, so long as the actual address is enclosed in angle brackets (<>).

Some examples of RFC-822 header lines include the following

```
From: bill@bloom-county.outland.com (Bill the Cat)

To: “Wayne Gretzky” <gretzky@la-kings.nhl.org>

From: MCP@TRON.NET (Master Control Program v1.0)

To: larry@startup.com (formerly larry@bigcorp.com)
```

Note that in the example the address in parentheses () is treated as a comment and will not be used for delivery.

The domain-part of RFC-822 addresses is case-insensitive; “user@bigcorp.com”, “user@BIGCORP.COM”, and “user@BigCorp.Com” are equivalent. The local-part of the address, since it is interpreted by a wide set of different local delivery agents on different operating systems, *is* case-sensitive, and should never be modified by mail transport programs.

UUCP Addresses

UUCP is a set of protocols intended to serve as a method of transporting files, executing remote commands, and transporting mail among UNIX systems. This was used to build the first large-scale electronic mail network as well as the Usenet News network. It is very commonly seen in the UNIX world, and there are implementations of UUCP for DOS/Windows systems and other platforms. Many Internet sites also use UUCP, and messages from one UUCP site to another may use the Internet as a backbone network.

UUCP mail addresses are of the form “site!user”, where “site” is the UUCP name of a particular system, and “user” is a mailbox. A chain of site names, called a “path”, can be used to explicitly state the route to the recipient’s site. e.g., *violet!topaz!ruby!jane* means, “send the message to *violet*, then *topaz*, then *ruby*, which will deliver it to mailbox “jane.” The namespace of UUCP sites is flat, and there is no provision for centrally assigned site names. There is, however, a central registry that attempts to both prevent name collisions and provide a routing database. The UUCP mail network is implemented as a number of individual, site-to-site connections.

In recent years there have been some modifications and extensions to UUCP mail addressing and routing. First, many UUCP sites have registered DNS domain names, and use normal RFC-822/DNS addresses. In most cases, these

sites have an Internet mail exchanger that accepts mail for the domain and forwards it via UUCP. Outgoing messages are sent to a default UUCP connection for delivery to the Internet or elsewhere.

Furthermore, it is possible to use the data provided in the centrally maintained UUCP routing/site maps to produce a routing database that will look up an address like *ruby!jane* and turn it into the appropriate path address. e.g. *violet!topaz!ruby!jane*, or alternatively forward the message to an Internet forwarder that may transform the address into a domain-type address like *jane@ruby.xyz.edu*.

Finally, if a site is in the UUCP map database, it may be possible to route mail to it by using the unofficial ".uucp" pseudo-domain, e.g., *jane@ruby.uucp*. This is functionally equivalent to *ruby!jane*, and requires that the sending system either use the UUCP map database or forward all outgoing messages to a mail relay system that does.

Hybrid Addresses

Because of the way certain sites are connected to mail relays, it is not uncommon to see addresses that mix different addressing formats. The most common of these is the so-called "percent-sign hack", in which a non-Internet mail route is "hidden" inside the local-part of an RFC-822 domain-type address. Instead of a simple mailbox, the local-part would contain an unofficial "user@site.network" type address. However, it is a syntax error for there to be more than one "@" in an RFC-822 address, so the "hidden" @ is turned into a "%", e.g., *jane%ruby.uucp@relay.xyz.edu*. In this case, the message will be delivered to *relay.xyz.edu*, whose mailer is expected to decode the local-part and deliver it on to *ruby!jane* by UUCP .

Another type of hybrid address is a mixture of a UUCP path address with a domain address, e.g., *ruby!jane@topaz.xyz.edu*. This is commonly seen, but has the problem of being ambiguous. It could mean either, "deliver the message to *topaz.xyz.edu* for forwarding to jane at the UUCP site *ruby*", or "deliver the message to UUCP site *ruby* for forwarding to jane at the domain *topaz.xyz.edu*." Unfortunately, there is no universal standard that covers these situations: Internet hosts will generally follow the former interpretation ("@" takes precedence) while UUCP-only hosts will follow the latter interpretation. This type of address format should be avoided wherever possible in favor of either a true domain address or, if necessary, a UUCP path address including the host in the domain-part, e.g., *topaz.xyz.edu!ruby!jane*.

Lastly, it is possible to encapsulate other types of address formats completely unrelated to RFC-822 or UUCP inside the local-part of a domain address. In this case, the entire local-part should be enclosed in double quotes (""). This can be useful for sending messages via X.400 gateways, e.g.,

```
"/C=US/PRMD=StarMail/O=HiTechCorp/G=Mickey/S=Jones/"@hitech.com
```

which indicates that the message is to be delivered to "hitech.com" via normal Internet methods (i.e., SMTP). The mail transport agent at hitech.com will then hand it off to a X.400 gateway for further delivery using the local-part.

Internet Mail Routing

With a properly formed domain address, the task of mail routing is relatively simple. When the sender's mail transport agent attempts to deliver the message on the Internet, it first "resolves" the address. After determining that the address is not local (in which case it will simply pass the message to a local delivery agent), it makes a DNS query asking for the host name and Internet address of a mail exchanger (MX) for the recipient's domain.

A mail exchanger is an Internet host that is registered as accepting mail for a particular host (including itself) or entire domain. When queried for an MX record for a domain, the DNS server will return records containing the host names and IP addresses of the mail forwarders and a priority number indicating which forwarder is to be tried first. The sending system will attempt to open an SMTP connection with the MX host with lowest-numbered priority, and if unsuccessful, will try each other MX in order of increasing numerical priority.

If there are no MX records for a particular host/domain, then the sending system will make a DNS query for the actual IP address of the recipient's host, and if successful, will attempt to open an SMTP connection directly with it.

MX records are typically used for two purposes. First, to provide a back-up capability for receipt of mail when a host is unavailable on the network. Secondly, when a host acts as a mail relay for systems that are not directly connected to the Internet.

In the first case, incoming messages may be re-directed to another host at the same site. Here, the messages might be deliverable by a different method, or at least could be queued locally until the original host is back on the network.

In the second case, a host acting as a mail relay will accept messages for one or more systems or sites that are not directly connected. It may either simply send the message on through an internal IP route (as in the case of systems behind security fire walls), or may forward the message onward by another type of transport, such as UUCP. In either case, the complexities of routing the mail are hidden from the sender of the message, who merely needs to know the proper domain address.

CHAPTER 3
APPLE MACINTOSH FILE STRUCTURE

The filesystem used by MacOS (the operating system running on Apple Macintosh computers) has a different file structure than MSDOS, UNIX or most other operating systems. On the latter, files can be seen as simple sequential unstructured streams of bytes, while Macintosh files consist of three parts:

- A *header*, containing information about the file (e.g., *true* filename, identity of the application that created and can subsequently open the file, file type (*TEXT* or more specific qualifiers), creation and last modification time, etc.

- Optionally, a *Resource fork* containing Macintosh-specific data (such as icons and other resources)

- Optionally, a *Data fork* containing the actual file data.

These three parts may be stored together in *AppleSingle* format, introduced in 1990 for Apple's version of UNIX called A/UX, and now considered Apple's official export format to other platforms. This is a sequential byte stream that can be passed to other operating systems (like DOS or UNIX). However, its direct use by applications on those platforms can be difficult, because the header and data fork are generally seen by the application as unrecognizable data. It is customary to discard the header and resource fork and pass only the data fork to applications running on non-Macintosh platforms.

Other binary formats similar in concept to AppleSingle are MacBinary I and its successor MacBinary II, understood by some FTP servers but not necessary for **Internet Exchange**.

Shipping Apple files over non-eight-bit-clean channels

Another way to handle Macintosh files in foreign environments is to convert them into a sequence of printable ASCII characters. This operation is especially important if the files have to be transmitted over a 7-bit communications channel, such as the old X.25 links or the SMTP Internet mail protocol. Various solutions can be chosen: UUENCODEing an AppleSingle stream, using the BinHex encoding format made popular by the *StuffIt* utility, or preferably the MacMIME standard as specified by the Internet document RFC1740 (for MIME AppleSingle and AppleDouble) and RFC1741 (for MIME Binhex).

MIME AppleSingle is simply a base64 (or otherwise) encoded AppleSingle binary, labeled as *Content-Type: application/applefile.* AppleDouble is a more flexible example of the MIME *Multipart* content type. It consists of a pair of MIME bodyparts, the first being an AppleSingle file derived from the original minus the data fork, and the second an application-specific MIME type (e.g.,

image/gif) containing the data fork information. That allows easy separation of the data fork on non-Macintosh platforms, at the same time preserving the Macintosh-specific information.

cc:Mail and Macintosh files

cc:Mail can store Macintosh files in the Post Office in AppleSingle binary format. Clients running on various platforms will get the complete file or just the Data fork. The cc:Mail client for Windows automatically strips away header and resource fork, whereas the client for Macintosh makes good use of the complete file. On the other hand, when the Macintosh client processes a non-AppleSingle attachment, it creates a dummy header and an empty Resource fork. The resulting document may or may not be readable by the original application running on the Macintosh. For example, Microsoft Excel® 5 does not recognize these rebuilt files, but Microsoft Excel 6 does and opens them normally.

Internet Exchange Macintosh file handling

From the above discussion, it is generally advisable to preserve as much file information as possible. This is the default behavior of **Internet Exchange**: outgoing cc:Mail AppleSingle attachments are encoded in MacMime or BinHex formats, and non-AppleSingle attachments are encoded as simple MIME or uuencoded. The same happens for incoming messages: MacMime and BinHex produce AppleSingle cc:Mail attachments, and the other formats produce native cc:Mail attachments.

There are situations when the administrator may want to modify this behavior. For example, if many of the cc:Mail clients fed by the gateway are not Macintoshes, storing the header and resource fork wastes disk space. Conversely, heavily Macintosh-oriented sites may want to synthesize AppleSingle cc:Mail attachments out of non-Macintosh MIME messages, using parameters (Creator and Type) determined by the MIME Content-Type headers. This might be preferable to delegating the job to the cc:Mail client when the mail is read, by which time the information contained in the MIME headers is lost. This may make the difference between being able to open a spreadsheet with Excel 5 or being stuck with an unreadable document.

To allow this choice, there are two mutually exclusive options in the *Configure Options* screen. The first is *Force Native* that strips the header and Resource fork from incoming MacMIME or BinHex messages and imports the messages as native, i.e. non-Macintosh. The alternative is *Force Apple* which creates an empty Resource fork and a synthetic header (based on the MIME headers) for non-Macintosh MIME or non-BinHex incoming messages, and imports the messages in AppleSingle format. For outgoing mail, the same two options apply to each peer in the *Configure Peer* screen.

In summary:

- If primarily running Macintoshes and/or there is plenty of free disk space where the local post office resides, set *Force Apple* in the *Configure Options* screen, otherwise set *Force Native.*

- If a certain domain has mainly PCs or UNIX boxes, and/or wants to eliminate unnecessary traffic, select *Force Native* in the *Configure Peer* screen for that domain.

- If a certain domain has mainly Macintoshes, select *Force Apple* in the *Configure Peer* screen for the corresponding domain.

- In all other case, stick to the defaults.

Where To Find More Information

Macintosh file types

The manual for the *Fetch* utility:

http://www.dartmouth.edu/pages/softdev/fetchhelp/index.html

Information about BinHex and pointers to related documents:

http://www.natural-innovations.com/boo/binhex.html

Macintosh and MIME RFCs

MIME Encapsulation of Macintosh files - MacMIME - also contains information on AppleSingle and AppleDouble formats:

http://www.internic.net/rfc/rfc1740.txt

MIME Content Type for BinHex Encoded Files:

http://www.internic.net/rfc/rfc1741.txt

Tools For General Users

StuffIt utility to handle BinHex files, by Aladdin Systems:

http://www.aladdinsys.com/obstufex.htm

Tools For Hardcore Hackers Only

UNIX C program for converting a file created by BinHex (usually named with one of the extensions ".hex", ".hcx", or ".hqx") into three host-system files (with the extensions ".info", ".data", and ".rsrc") containing respectively header, data fork and resource fork. Great for learning about the gory details, but not much use to most administrators.

ftp://oak.oakland.edu/SimTel/msdos/Macintosh/xbin23.zip

CHAPTER 4
ATTACHMENT NAMING CONVENTIONS

Internet Exchange uses several different methods to communicate attachment names in outgoing messages. This allows the names of cc:Mail attachments to be preserved when being sent to users on the Internet. In a similar manner, the same methods are used to extract attachment names from incoming Internet messages.

The Name parameter

This parameter is optional in the first MIME specification RFC-1341 and is deprecated by RFC1521, but is still used by many mail programs. It is generated by **Internet Exchange** by checking the MIME configuration information available through the *Configure MIME* screen. The parameter is appended to the MIME *content-type* header. For example:

Content-Type: image/gif; name="world.gif"

For incoming messages, any present name parameter is used to name the corresponding attachment when imported into cc:Mail. If absent, a unique filename *MIMEnn.ext* will be generated, *nn* being a small decimal number, and *ext* an extension determined by the MIME mappings or, if these do not help, the default extension *raw*. For example:

Content-Type: application/octet-stream; name="mime01.raw"

The content-disposition header

This header is defined in RFC1806 and is an alternative way of defining attachment names. For example:

Content-Disposition : attachment; filename="test.doc"

This specifies that the MIME bodypart is an attachment, as well as its original filename. For incoming messages, this information is again used to generate a name for cc:Mail attachments.

Macintosh header information

For incoming attachments in either AppleSingle or BinHex format, the attachment name will be taken from the Macintosh header resource. This takes precedence over either of the above methods of attachment naming.

PART 2 INSTALLATION

CHAPTER 5
PLANNING YOUR INSTALLATION

INSTALLATION REQUIREMENTS

Before you can install **Internet Exchange**, all the hardware and software components listed on pages 2 - 3 need to be properly installed and running.

VIM LIBRARY INSTALLATION

Internet Exchange requires Lotus cc:Mail VIM version 2.07 or later to be present on the system in order to properly operate. At the time of this writing, the cc:Mail VIM libraries are up to version 2.21. If you are using a recent version of cc:Mail for Windows, you probably have the most recent versions of the VIM libraries. To verify that you are running version 2.21, check the following VIM libraries against the versions you have installed. This can be done by running the *DIR* command from the Windows directory.

ccedit.dll	98800	10-11-95
ccsmi.dll	70144	10-11-95
ccutil.dll	102720	10-11-95
cdvim.dll	47856	10-11-95
charset.dll	9216	10-11-95
maileng.dll	371712	10-11-95
memman.dll	10240	10-11-95
smi.dll	36864	10-11-95
vim.dll	156672	10-11-95

If the files match the ones above or are more recent, then you are running with at least version 2.21 of the VIM libraries and do not need to do anything more. If they are not as recent as the above files, you can contact cc:Mail technical support to request the proper libraries. Alternatively, you can download them from the Lotus BBS, or you can obtain them by anonymous FTP over the Internet. The cc:Mail BBS can be reached at:

BBS: (415) 691-0401 (Call with any asynchronous package;
parameters: 8-N-1)

To obtain the files by anonymous FTP, the current URL:

ftp://ftp.ccmail.com/pub/comm/ccmail/dev_tools/vdlwin.zip.

INTERNET EXCHANGE INSTALLATION

Before you run the **Internet Exchange** *INSTALL* program, it is necessary to assemble all the information needed for the installation. This includes information about where to install **Internet Exchange**, and details about the cc:Mail gateway post office as well as your TCP/IP configuration.

To assist with the installation of the gateway software, please review and fill out the installation worksheet on the following page prior to the start of the installation. Each item in the worksheet will be discussed in the following sections.

Gateway Parameters

The Gateway Parameters sections of the installation worksheet identify parameters that are associated with either the installation and/or the overall operation of the gateway.

Program Directory

The default location for **Internet Exchange** to be installed is c:\ieccmail. This is where the programs and libraries reside. This directory can be located anywhere, however it is strongly recommended that it be placed on a local hard drive for performance and reliability reasons.

Queue Directory

The default location for **Internet Exchange** to store temporary files is c:\ieccmail\queue. This is where the messages and log files reside. This directory can be located anywhere, however it is strongly recommended that it be placed on a local hard drive for performance and reliability reasons. However, the program and queue directories do not have to be on the same disk. This allows for a more flexible installation.

Temporary Directory

This is the directory in which **Internet Exchange** will store messages. It is usually configured to be a subdirectory of the queue directory. However, it can be set to a different directory and/or drive depending upon local disk availability.

Local Character Set

The ISO character set to be used. Most Anglo Saxon countries can select US-ASCII, while others will prefer to choose a different character set. All outgoing email will be tagged as using the selected character set.

Local Time Zone

This is the time zone in which the local machine resides. Whether this time zone uses daylight saving or not should also be noted. There are many locations configured in the system, including the USA, much of Europe, and Asia. If the local time zone is not listed, then it will have to be entered manually into IMA.INI with an editor as follows:

[Gateway]
Timezone=*tzn*[[+ | -]] *hh*[[:*mm*[[*:ss*]]]][[*dzn*]]

The *tzn* must be a three-letter time-zone name, such as PST, followed by an optionally signed number, *hh*, giving the difference in hours between UCT and local time. To specify the exact local time, the hours can be followed by minutes, *:mm*; seconds, *:ss*; and a three-letter daylight-saving-time zone, *dzn*, such as PDT. Separate hours, minutes, and seconds with colons (:). If daylight saving time is never in effect, as is the case in certain states and localities, set *Timezone* without a value, for *dzn*. If the *Timezone* value is not currently set, the default is PST8PDT, which corresponds to the Pacific time zone of the USA.

If the time zone "Use system TZ variable" is selected, the timezone information will be obtained from the user defined TZ environment variable. Under Windows 3.1 and Windows 95, this can be set in the *autoexec.bat* system startup file. Under Windows NT it is usually set in the system registry. In either case, the machine must be rebooted in order to make the change effective.

cc:Mail Parameters

The cc:Mail section of the installation worksheet identifies parameters associated with the gateway post office. This is the post office that is queuing messages on behalf of **Internet Exchange**.

Local Post Office Name

This is the name of the post office that **Internet Exchange** will log into to retrieve messages.

Internet Post Office Name

This is the name which **Internet Exchange** uses to log into the cc:Mail Post Office. This name must exist in the cc:Mail directory, and must be defined as a Post Office. Although any unique name may be used here, it is recommended that *Internet* be used for clarity.

Post Office Path

This is the path name for the directory where the local post office resides.

Post Office Password

This is the password that **Internet Exchange** should use when logging into the local post office.

Post Office Postmaster

Internet mail standards require that each site have a mail account that receives messages addressed to "postmaster." The postmaster typically receives notices about mail problems, network problems, and inquiries about users and mailboxes. It should be set to the cc:Mail address of the person responsible for the Internet mail gateway.

TCP/IP Parameters

The TCP/IP parameters section of the installation worksheet identifies parameters associated with the local TCP/IP network.

Host Name

Each host on the Internet must have a unique identifier so that email bound for that site has a single unambiguous destination. This identifier is known as the Fully Qualified Domain Name (FQDN).

The host name parameter is the name component of the gateway machine FQDN. For example, if the FQDN of the gateway is *iegate.jade.net* then the host name would simply be *iegate*.

Domain Name

This is the domain component of the gateway machine FQDN. For example, if the FQDN of the gateway is *iegate.jade.net* then the domain component would be *jade.net*.

Host File Location

This is the full path name of the TCP/IP host file. Even if the Domain Name System (DNS) is used for host name to address translation, it is recommended that a host file be present that contains addresses for the following hosts: loopback, gateway machine, and your mail relay host.

Mail Relay Host

The mail relay host is the name of the machine that is used to send mail that cannot be resolved by either host table lookup or by DNS queries (or if you have configured **Internet Exchange** to use a default mail relay host only). It is required that this host have an entry in the local host table in the event that the DNS cannot be contacted.

Name Server Addresses

This is the list of name servers to contact for performing MX record and Address record lookups using the Domain Name System (DNS).

Internet Exchange

Installation Worksheet

Gateway Parameters

Program Directory _____

Queue Directory _____

Temporary Directory _____

Local Character Set _____

Local Time Zone _____

cc:Mail

Local Post Office Name _____

Internet Post Office Name _____

Post Office Path _____

Post Office Password _____

Post Office Postmaster _____

TCP/IP

Host Name _____

Domain Name _____

Host File Location _____

Mail Relay Host _____

Name Server Addresses _____

CHAPTER 6
INTERNET EXCHANGE INSTALLATION

PREPARING THE CC:MAIL POST OFFICE

Internet Exchange requires that a gateway post office be established within cc:Mail. The following instructions provide a step-by-step approach to the creation of this gateway post office.

1) Start up cc:Mail's **admin** utility. First enter the DOS environment, change your current working directory to the directory containing the above program, e.g., c:\ccadmin, and then type **admin** followed by **Enter**.

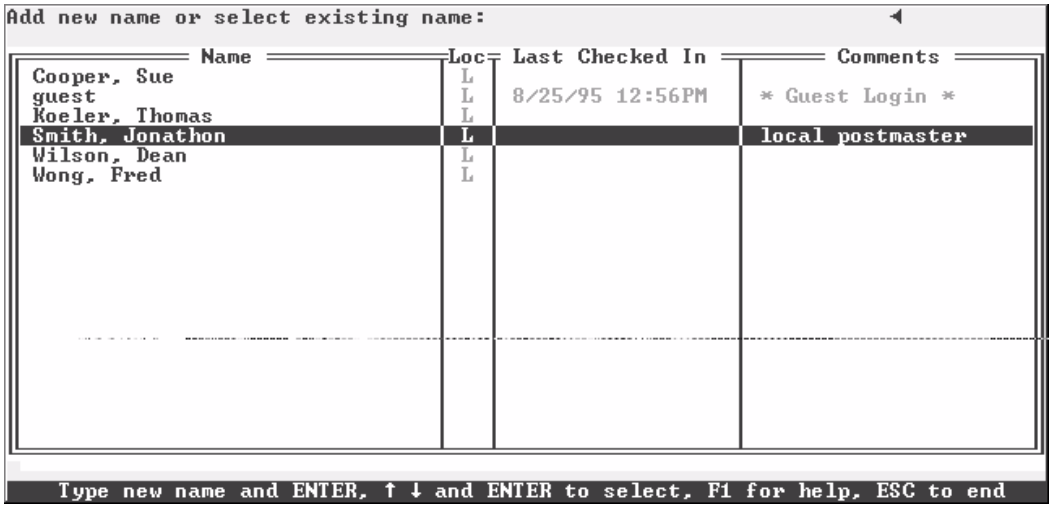
2) Enter the local post office directory, e.g. c:\ccdata, and press **Enter**.

3) Enter the name of the local post office - in our example, *Jade Networks*, and press **Enter**.

4) Enter the password for the post office and press **Enter**.

The following screen will be presented:

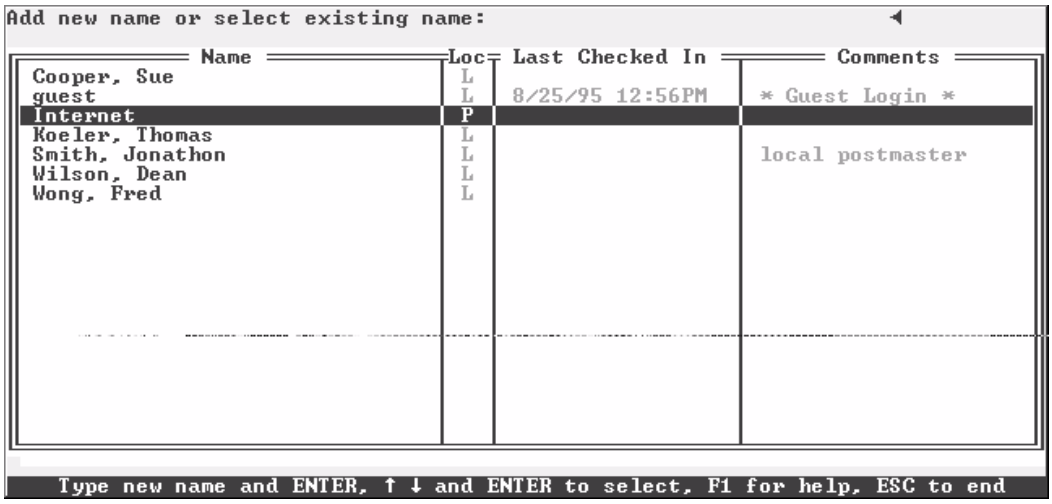
5) Choose “manage Mail directory” (as above) by selecting and pressing **Enter**.



6) Type in the name of the new gateway post office, e.g., Internet, and press **Enter**.

7) When prompted for the location status of the new post office enter a "**P**" followed by **Enter**.

8) You can now enter an optional comment (e.g., "Internet Exchange post office"), followed by **Enter**. The new entry now appears in the menu:



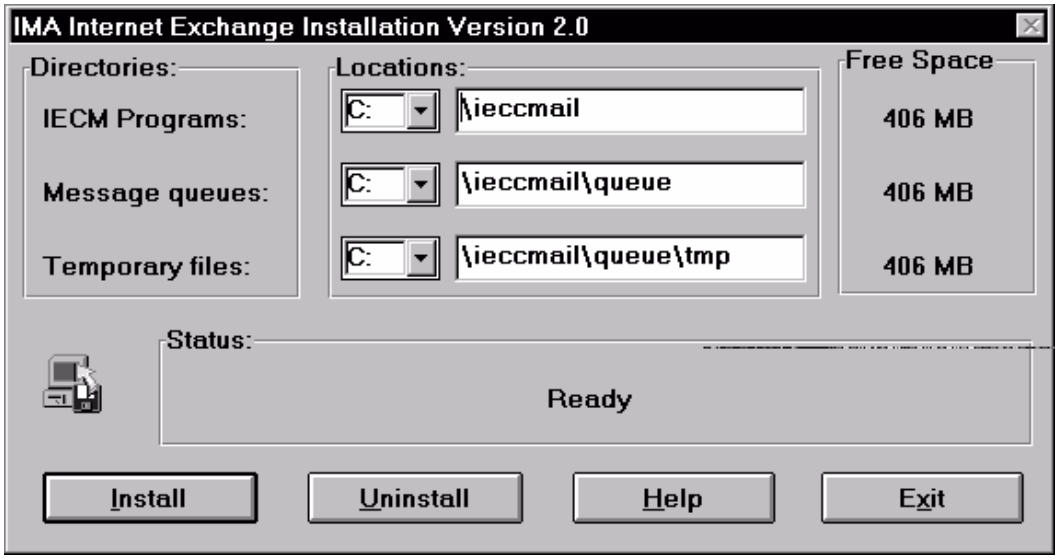
10) After verifying that the above information is correct press **Enter** followed by **Esc** to return to the main menu, then enter an **X** followed by **Enter** to save and exit **admin**.

INSTALLING INTERNET EXCHANGE

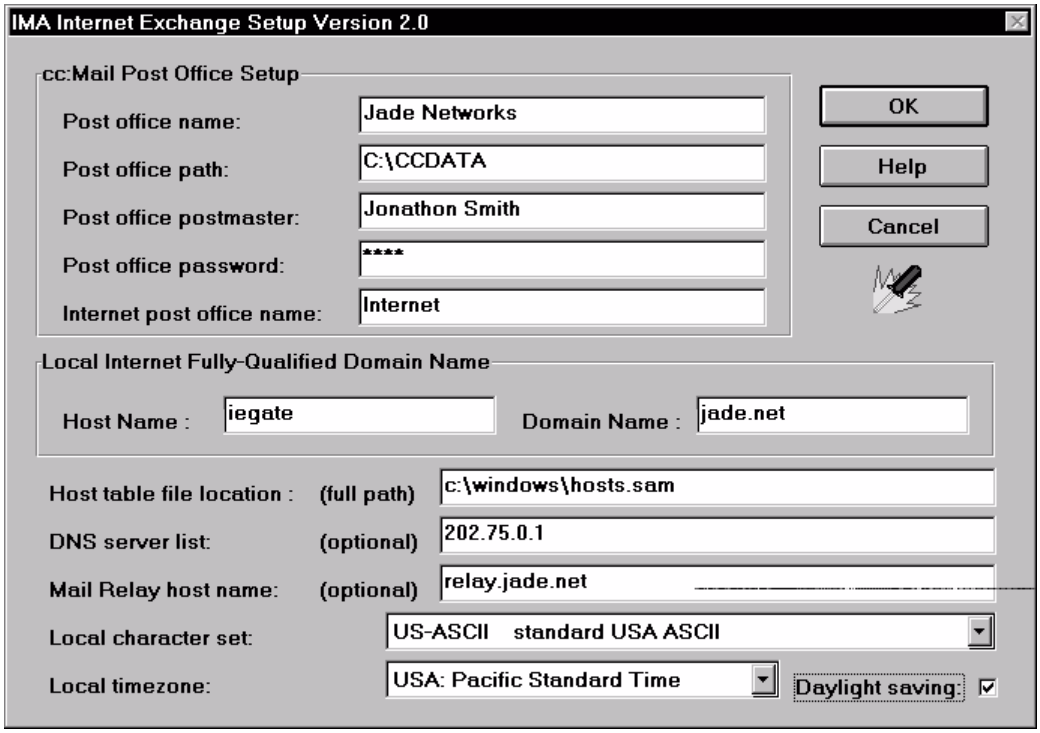
Once the Installation Worksheet is complete, you are ready to perform the actual installation. To install **Internet Exchange**, perform the following tasks:

1. Make sure your computer and monitor are turned on and the Windows environment running.
2. Insert the diskette labeled ***Internet Exchange*** - *Program Disk* into an available diskette drive.

3. Choose *Run* from the Program Manager File menu.
4. Run the program **install** located on the *Internet Exchange Program* Disk.
5. **Install** will then present the following screen:



Press the default *Install* button to install **Internet Exchange**. It will display a progress dialog showing how the installation process is going. Once all the files have been copied over and the **Internet Exchange** Program Group has been created, the *Setup* program will be automatically executed and will display the following screen:



This screen shows the basic parameters that are necessary to start up **Internet Exchange**. The above example is for the machine *iegate.jade.net*, connecting to the cc:Mail post office *Jade Networks* located in c:\ccdata. The Internet cc:Mail Post Office name is *Internet.* Please refer to the Installation Worksheet completed earlier, and ensure that all entries above are properly set for your site. When all fields are correct, click on the *OK* button for the selections to take effect.

If **Internet Exchange** is being installed on top of an existing installation, it may be necessary to perform certain conversions from the previous version. If necessary, the following utilities will be automatically started during the installation process: *MIME Magic Mapping Utility*, *Message Conversion Utility*, *Domain Conversion Utility*, and the *Address Conversion Utility*. Detailed information regarding each of these utilities can be found in Chapter 10.

Once *Setup* is complete, it will offer to update the gateway license. If desired, the *License Update* program will be automatically started. This program can also be run at any time by selecting the *License Update* program from the *Internet Exchange for cc:Mail* Program Group.

After License Update is started, the following screen will be displayed:

There are two read only fields that appear when the *License Update* program starts - the FQDN and Serial Number. The FQDN, or fully qualified domain name, is the official name for the gateway. The Serial Number field displays the unique serial number built into the current copy of *Internet Exchange.* These are the two fields that need to be provided to your supplier or IMA to generate a license key as described below.

There are three types of software licenses that can be generated for *Internet Exchange.* They are:

Evaluation License
If this copy of *Internet Exchange* was obtained from a public access site (anonymous FTP, the Web or a commercial service), it is not possible to permanently enable the gateway. For these versions of the gateway, the evaluation license radio button will be the only option.

To obtain an evaluation license, it is necessary to contact your supplier or IMA, either by phone, fax, or email. Email requests should be sent to:

eval-auth@ima.com

When requesting an evaluation license, please provide a completed registration form, which can be found in the Internet Exchange for cc:Mail program group.

After obtaining the evaluation license, it is necessary to enter both the expiration date of the license as well as the license key. The expiration date is entered in the form mm/dd/yy. The license key should be entered exactly as obtained from your supplier or IMA. After entering the appropriate expiration date and license key, press the *Update* button to store the registration information.

Interim License
Interim licenses are similar to Evaluation licenses, with the exception that Interim licenses can be updated in the field to a Permanent license at a later date. To obtain and apply an Interim license, follow the same procedure as outlined for an Evaluation license. Instead of sending email to *eval-auth*, please contact either your supplier or IMA, or send email containing full registration information to:

auth@ima.com

Permanent License
Unlike Evaluation and Interim licenses, Permanent licenses do not have any expiration date associated with them. These licenses are based upon the **Internet Exchange** serial number and the Fully Qualified Domain Name (FQDN) of the gateway machine. To obtain a Permanent license key, please contact either your supplier or IMA, or send email containing full registration information to:

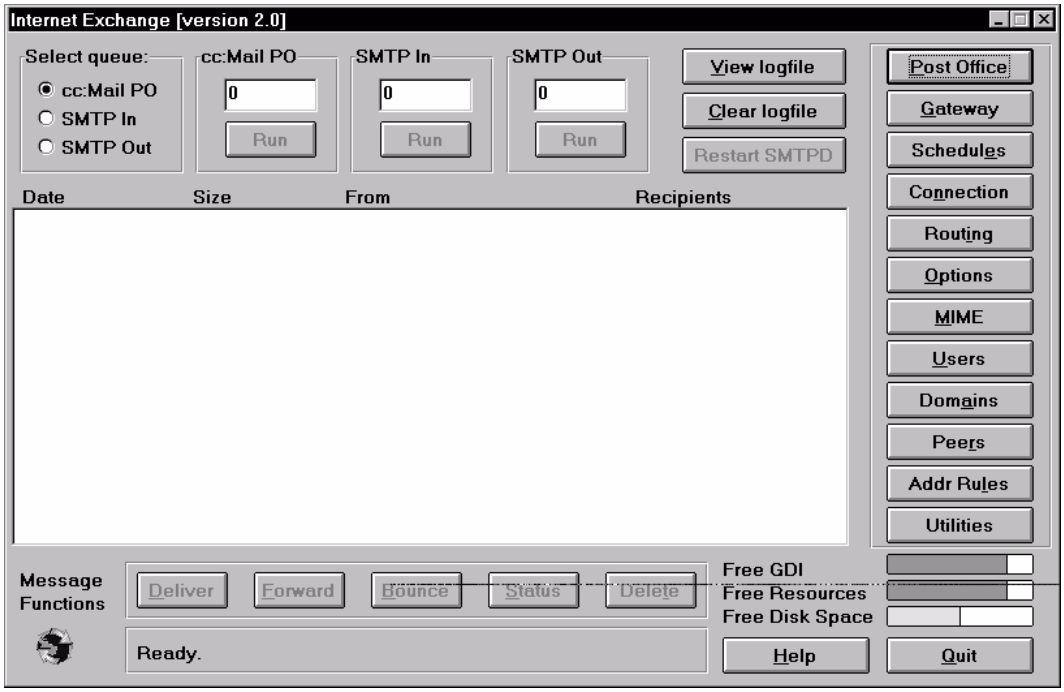
auth@ima.com

Since there is no termination date to a permanent license, the only information you need to enter for a Permanent license is the license key. Enter the key and press the *Update* button to store the registration information.

After storing the license information with License Update, the installation and licensing of **Internet Exchange** is complete and the gateway is ready for use. You may wish to perform further configuration of the users, subdomains or the MIME Mapping Database.

STARTING INTERNET EXCHANGE

During the **Internet Exchange** installation process, the program group ***Internet Exchange*** is created and the icons for the following programs are set up: *SYSMAN*, *CCIN*, *CCOUT*, *SMTPD*, and *SMTPC*. To start **Internet Exchange**, double click on the System Manager icon with the title *IMA Internet Exchange*. The system manager will then start, along with the SMTP Daemon and the following window displayed:



The rotating globe in the lower left hand corner of the System Manager indicates normal operation of the gateway, although its rate of rotation can vary. In addition to putting up the System Manager window, the SMTP daemon icon should be visible in the lower left hand corner of the display. The existence of the following icon indicates that the SMTP Daemon is running and listening for incoming SMTP requests:

If the SMTP daemon should fail for any reason, the icon will disappear, and the *Restart SMTP* button on the System Manager will become activated.

At this point, **Internet Exchange** is up and running, waiting to exchange messages between cc:Mail and the Internet.

PART 3 OPERATION

CHAPTER 7
CONFIGURING INTERNET EXCHANGE

CONFIGURATION OPTIONS

Normal configuration of **Internet Exchange** is done while the gateway is running. For an optional manual setup of configuration options, see the file *iecmref.pdf* in the **Internet Exchange** installation directory.

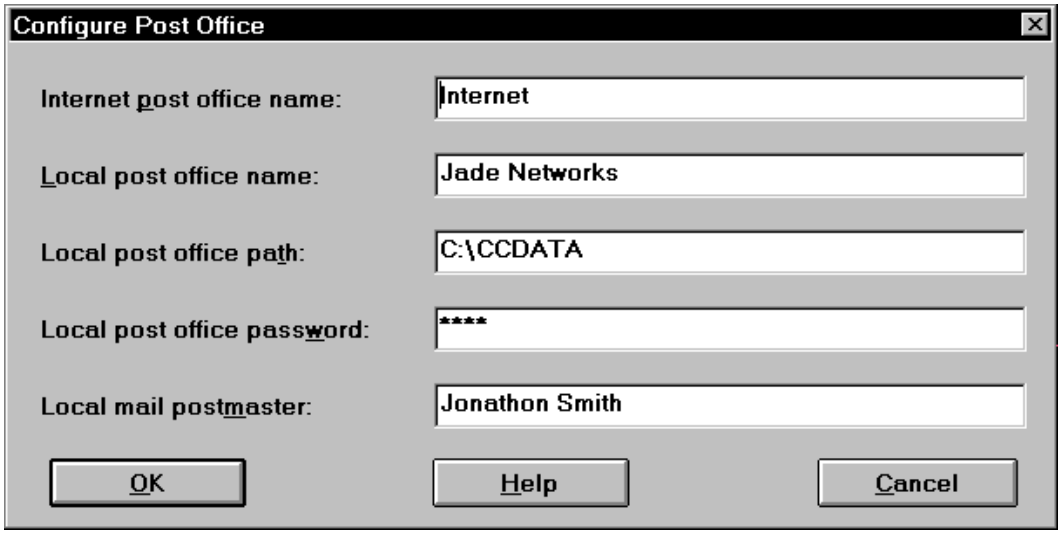
The configuration of **Internet Exchange** is broken down into several functional groups. These are identified by the buttons on the right hand side of the System Manager window and are made up of the following:

Post Office	Configures the name and location of the local cc:Mail post office. Also stores the local post office password and identifies the postmaster.
Gateway	Configures general gateway parameters including mode of operation, logging levels, system maximum sizes, and queue directory location.
Schedules	Configures the frequency at which the various queue managers are launched.
Connection	Configures the host name, domain name, and location of the host table, and alternate names by which the gateway may be known by.
Routing	Configures the list of DNS hosts to check with for name resolution. Also identifies the name of the mail relay host(s) for default mail routing.
Options	Configures the default MIME encoding method, the default addressing separator as well as various options used in message delivery to cc:Mail.
MIME	Defines the mapping between DOS file extensions and MIME Content-Type and Content-Subtype identifiers.
Users	Defines special cc:Mail username to Internet address mappings.
Domains	Defines mappings between cc:Mail Post Offices and Internet subdomains

Peer	Defines capabilities of peer systems with which the gateway communicates.
Addr Rules	Defines addressing rules for simplifying conversion between cc:Mail usernames and Internet addresses.
Utilities	Allows execution of several file utilities for maintaining various gateway databases.

CONFIGURE POST OFFICE

This screen allows entry of the required information to allow **Internet Exchange** to communicate with the cc:Mail post office.



Internet Post Office Name

This is the name which **Internet Exchange** uses to log into the cc:Mail Post Office. This name must exist in the cc:Mail directory, and must be defined as a Post Office. Although any unique name may be used here, it is recommended that *Internet* be used for clarity. **Internet Exchange** uses this information to access the local cc:Mail Post Office.

Local Post Office Name

This is the name of the local cc:Mail Post Office. This will be the same as the name parameter given to the *chkstat* and *reclaim* programs for cc:Mail. **Internet Exchange** uses this information to resolve local email addresses.

Local Post Office Path

This is the path where the local cc:Mail Post Office resides. This will be the same as the path parameter given to the *chkstat* and *reclaim* programs for cc:Mail. **Internet Exchange** uses this information to access the local cc:Mail Post Office.

Local Post Office Password

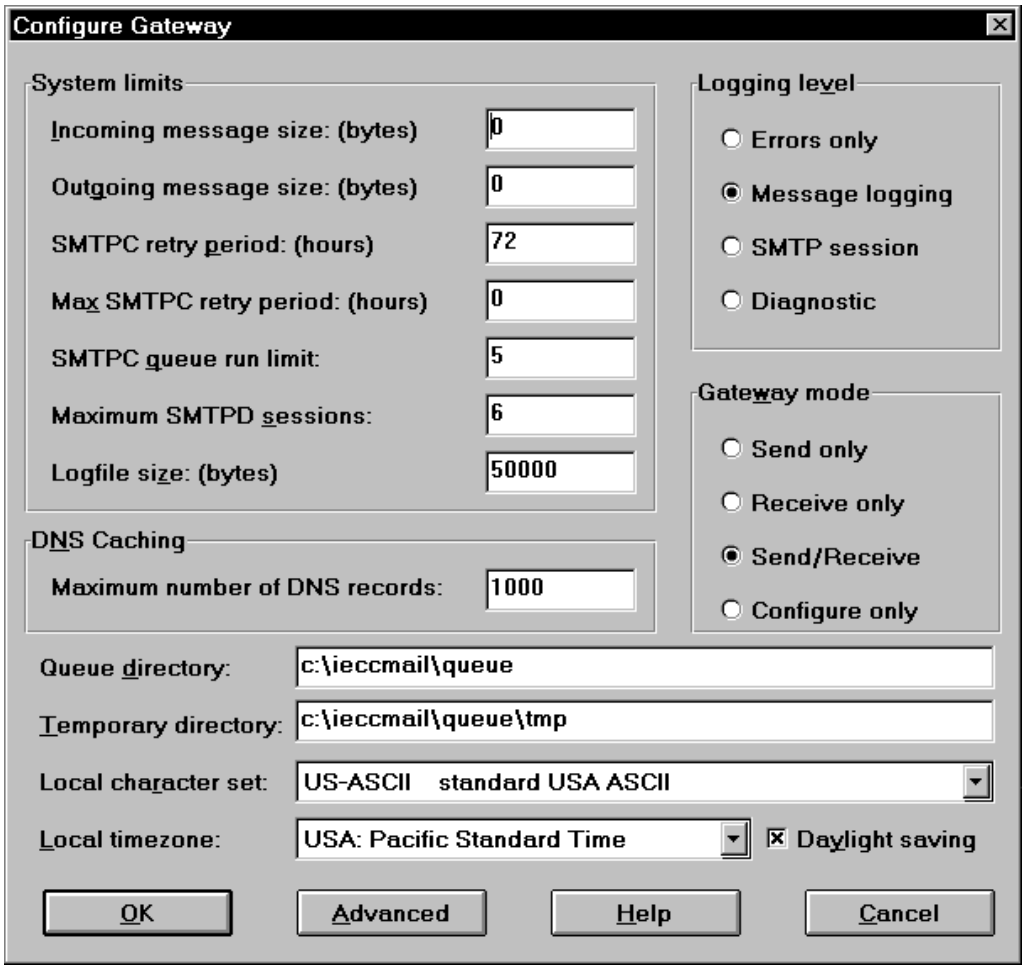
This is the password for the local cc:Mail Post Office. This will be the same as the password parameter given to the *chkstat* and *reclaim* programs for cc:Mail. **Internet Exchange** uses this information to access the local cc:Mail Post Office.

Local Mail Postmaster

This is the alias for the Internet Postmaster name. All Internet sites are required to support the Postmaster alias. When **Internet Exchange** receives a message addressed to Postmaster, it will be sent to this cc:Mail user.

CONFIGURE GATEWAY

This screen allows entry of information related to the general operation of **Internet Exchange**.



System Limits

Incoming Message Size:	The largest Internet message that the gateway will accept. Setting this field to the default of zero will disable any limits on the size of incoming messages.
Outgoing Message Size:	The largest cc:Mail message that can be sent out. Setting this field to the default of zero will disable any limits for outgoing messages.
SMTPC Retry Period :	The number of hours for SMTPC to try to deliver a message before returning it to sender. The default is 72 hours.
Maximum SMTPC Retry Period :	The maximum number of hours before SMTPC attempts to deliver a delayed message. SMTPC uses an exponential backoff algorithm to avoid retrying a host too often. Each retry is attempted after a delay twice as long as the last delay. This parameter puts a limit on this delay, so that it does not get too large. The default is zero which indicates no maximum.
SMTPC Queue Run Limit :	Whenever possible, SMTPC attempts to deliver as many messages as possible in a given SMTP connection. In order to do this, before establishing any connections, SMTPC performs an analysis of the outbound SMTP queue to determine which machines it needs to contact in a given queue run. When this value is set to 0, all messages in the queue are processed before any work is done. For very large queues, this process can be quite time consuming. Set this parameter between 5 and 20 for optimal performance. A value of zero to indicate no limit. The default is 5.
Maximum SMTPD Sessions :	Some stacks have trouble when too many incoming SMTPD sessions are active. This parameter limits the number of incoming simultaneous sessions. The default is 5 and maximum is 40. A value of zero to indicate no limit.
Logfile size:	The largest log file size allowed before the gateway saves it to another name and starts a new one. The default limit is 50,000 bytes which allows the Windows *notepad* application to read the file. A value of zero to indicate no limit.

DNS Caching

This entry records the maximum number of DNS records cached. The DNS cache greatly improves throughput of Internet Exchange, particularly when the DNS server(s) are not on a local LAN. The default is 1000, which balances increased throughput against greater disk space used for the cache.

Logging Level

Internet Exchange offers four levels of debugging:

Errors Only	Only errors will be logged.
Message Logging	Information about the delivery of each message will be logged.
SMTP Session	All SMTP conversations will be logged.
Diagnostic	A great deal of extra debugging data will be logged.

The logging levels are cumulative. e.g. at message logging level, errors are also logged.

Gateway Mode

Internet Exchange can operate in either send mode, receive mode, send/receive mode, or configure only mode. Send/Receive mode is the normal method of operation. Configure only mode can be used to completely shut down all gateway operations while configuration takes place. While this is not necessary, it may be desirable if gateway problems are identified. The radio buttons allow the gateway mode to be changed to any of these values.

Gateway Queue Directory

Internet Exchange will store messages in this directory. The log file is stored as *ieccmail.log*. The queue subdirectories are:

in:	messages coming in from the Internet.
out:	messages waiting to be delivered to the Internet.
bad:	directory for any malformed messages. These are sometimes created when a SMTP session doesn't finish cleanly.

The default is *c:\ieccmail\queue*

Gateway Temporary Directory

This is the directory in which **Internet Exchange** stores temporary files. Usually it is a subdirectory of the queue directory. It can be set to a different directory or drive depending upon local disk availability.

The default is *c:\ieccmail\queue\tmp*

Local Character Set

The ISO character set to be used. Most Anglo Saxon countries can select US-ASCII, while others will prefer to choose a different character set. All outgoing email will be tagged as using the selected character set.

Local time zone

This is the time zone in which the local machine resides. Whether this time zone uses daylight saving or not should also be noted. There are many locations configured in the system, including the USA, much of Europe, and Asia. If the local time zone is not listed, then it will have to be entered manually into IMA.INI with an editor as follows:

[Gateway]
Timezone=*tzn*[[+ | -]] *hh*[[:*mm*[[:*ss*]]]][[*dzn*]]

The *tzn* must be a three-letter time-zone name, such as PST, followed by an optionally signed number, *hh*, giving the difference in hours between UCT and local time. To specify the exact local time, the hours can be followed by minutes, *:mm*; seconds, *:ss*; and a three-letter daylight-saving-time zone, *dzn*, such as PDT. Separate hours, minutes, and seconds with colons (:). If daylight saving time is never in effect, as is the case in certain states and localities, set *Timezone* without a value, for *dzn*. If the *Timezone* value is not currently set, the default is PST8PDT, which corresponds to the Pacific time zone of the USA.

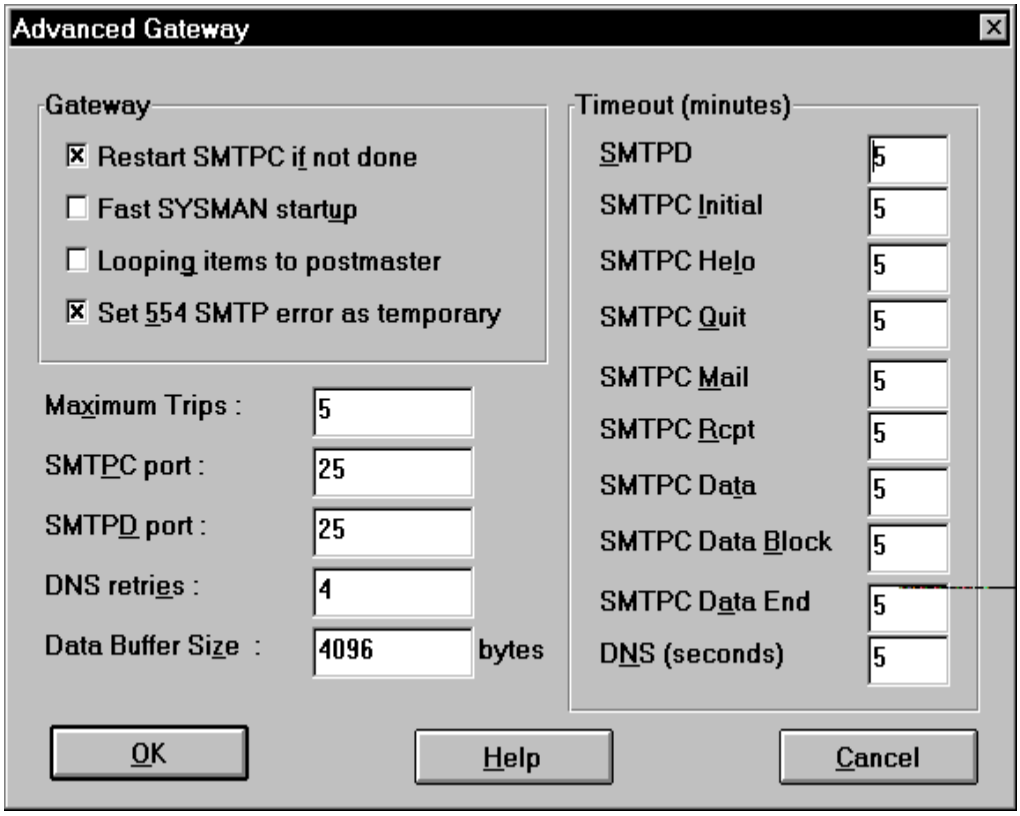
If the time zone "Use system TZ variable" is selected, the timezone information will be obtained from the user defined TZ environment variable. Under Windows 3.1 and Windows 95, this can be set in the *autoexec.bat* system startup file. Under Windows NT it is usually set in the system registry. In either case, the machine must be rebooted in order to make the change effective.

Daylight saving

This field is set to indicate whether the locale time zone uses daylight saving in the summer.

Advanced

This dialog box allows many of the advanced gateway parameters to be changed. Most of the time the defaults will work fine, and they should not be changed without fully understanding the consequences.



There are several functions that are controlled within the gateway section. They are:

Restart SMTPC if not done

If SMTPC does not finish the outgoing queue in one attempt, setting this option will restart it until all the outgoing messages have been delivered to the Internet. The default is YES.

Fast SYSMAN startup

If there has been a problem with the network, a large queue of messages can be built up in the cc:Mail Post Office. In that case, gateway startup will be very slow. By setting this variable, the queue counter update will not occur until either a key has been pressed or the mouse has been moved. Also, the queue messages display will not be updated until the next time a key is pressed or the mouse is moved. This will greatly increase the speed of gateway startup. This option should be set if running unattended dialup PPP. The default is NO.

Looping items to postmaster

If set, any looping messages will be routed to the local postmaster, instead of being returned to the remote sender. This is often useful to stop infinite email loops from occurring. The default is NO.

Set 554 SMTP error as temporary

RFC821 on SMTP is not clear as to whether the error *554 transaction failed* during the DATA phase should be regarded as a permanent error. Usually 5xx errors are permanent, but some SMTP servers return 554 errors for temporary errors. **Internet Exchange** takes the conservative approach and retries such messages later. If this option is set to NO, then such messages will be bounced instead of retried. The default is YES.

Timeouts

There are a large number of timeout values that can be altered if needed. The defaults of 5 minutes are usually adequate, and should be changed only if Internet Exchange is experiencing a lot of timeouts. The timeouts that can be changed are:

SMTPD	how long SMTPD waits on an open socket during a SMTP session.
SMTPC Initial	how long SMTPC waits for a reply when starting a new session.
SMTPC Helo	how long SMTPC waits for the remote system to respond to the HELO command.
SMTPC Mail	how long SMTPC waits for the remote system to respond to the MAIL command.
SMTPC Rcpt	how long SMTPC waits for the remote system to respond to the RCPT command.
SMTPC Data	how long SMTPC waits for the remote system to respond to the DATA command.
SMTPC Data Block	how long SMTPC waits for the remote system to acknowledge each block of data sent.
SMTPC Data End	how long SMTPC waits for the remote system to respond to the dot(.) command after all the data has been sent.
SMTPC Quit	how long SMTPC waits for the remote system to respond to the QUIT command.
DNS	how long to wait before a DNS request times out (default is 5 seconds).

Maximum Trips

This option specifies the maximum number of Received lines allowed in an incoming message that show the FQDN of the gateway machine. If this number is exceeded, the message will be bounced. This option is useful in preventing message loops. The default is 5.

SMTPC port

This option specifies the port which SMTPC uses, and can be useful when running the gateway behind a firewall, or any other non standard setup. The default is 25.

SMTPD port

This option specifies the port which SMTPD uses, and can be useful when running the gateway behind a firewall, or any other non standard setup. The default is 25.

DNS retries

This option specifies the number of times a DNS query is retried after a timeout. The default is 4.

Data Buffer Size

This is the size of the data buffer used by the SMTP programs to read data from the Internet. If the gateway machine uses disk caching, set this option to the size of the read ahead buffer. The maximum value is 32767, and the default is 4096(4kb).

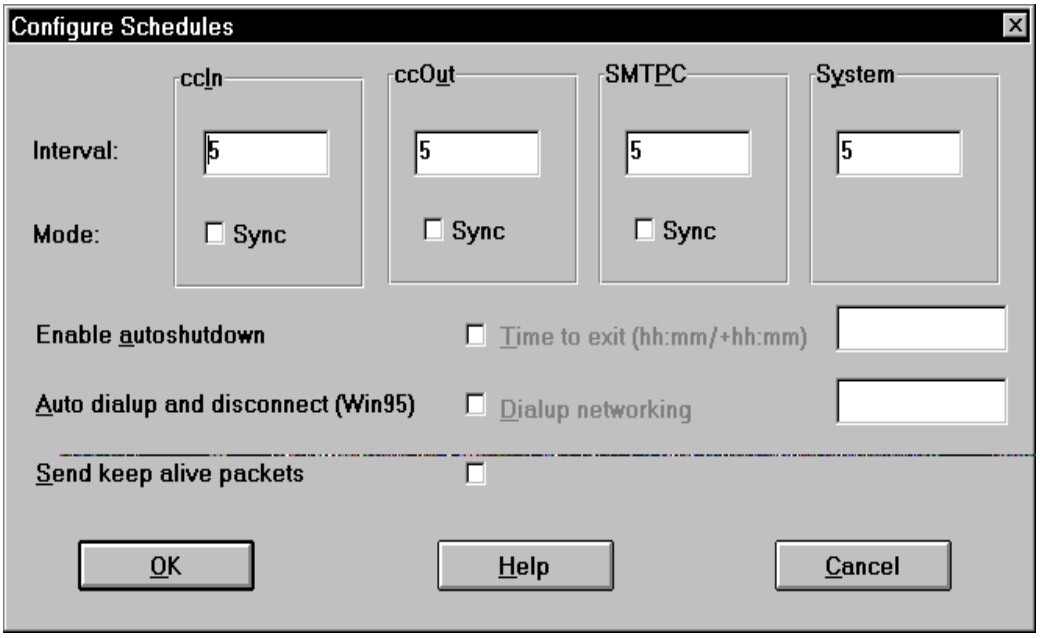
Low Disk Warning

This option *WarnIfSpaceLeft* is not configurable from within *SYSMAN*, and must be edited in the *Gateway* section of *IMA.INI* manually. It specifies the amount of free disk space (in MB) below which *SYSMAN* will issue a warning message in the status area of the main screen. This is an additional warning mechanism to the gauge indicating the amount of free disk space. The default is 5MB.

e.g. WarnIfSpaceLeft=5

CONFIGURE SCHEDULES

This screen allows entry of scheduling information for **Internet Exchange**:



CCIN Interval

The interval for starting up *CCIN*, measured in minutes. *CCIN* is the process responsible for the transfer of messages from the SMTP In queue into cc:Mail.

CCOUT Interval

The interval for starting up *CCOUT*, measured in minutes. *CCOUT* is the process responsible for the transfer of messages out of the cc:Mail post office into the SMTP Out queue.

SMTPC Interval

The interval for starting up *SMTPC*, measured in minutes. *SMTPC* is the process responsible for the transfer of messages from the SMTP Out queue to remote Internet hosts by SMTP.

System Interval

The interval for starting up system checking, measured in minutes. When the time interval is reached, Internet Exchange will check the system resources, such as free GDI resources and free USER resources. These are internal Windows structures relating to the user interface, windows and graphics which are crucial to the operation of the gateway. Note that USER resources are *not* related to either **Internet Exchange** users or cc:Mail users. It will also check on free disk space, time stamps of SMTP.ADR and SMTP.POD (if auto-conversion is enabled).

Sync Mode

Messages flow through **Internet Exchange** by way of the various different queue managers. For messages originating in cc:Mail, these managers are *CCOUT* and *SMTPC*. For messages originating outside cc:Mail, the corresponding managers are *SMTPD* and *CCIN*. Each of these queue managers operate independently of each other and are run by *SYSMAN* at regular intervals as outlined above.

Selecting the sync checkbox for each of the above will cause the corresponding queue manager to be started as soon as a message is available. For instance, if the CCIN Sync box is checked, as soon as a message is received by *SMTPD*, *CCIN* will be started to deliver the message to cc:Mail. For most installations, it is recommended to run with sync mode selected for all queue managers.

Enable Autoshutdown

This option will enable an automatic shutdown of **Internet Exchange** at the given time (24 hour time format). When set, the gateway will shut down at the requested time. However, if it is restarted during the same hour as it was shutdown, the gateway will again shutdown immediately. To avoid this, restart the gateway at a different hour than it was shutdown, e.g. shutdown at 2:30 and restart at 3:00.

A relative shutdown time can also be given, in the format of +hh:mm. This will shut the gateway down hh:mm from the time that *SYSMAN* was started.

Auto dialup and disconnect (Win95)

This option is only available when running on Windows 95. When checked, Internet Exchange will bring up the dialup networking section in Windows 95 and execute it automatically.

Dialup networking

The host to connect to for dialup networking. See the Windows help command for more information on setting up dialup networking, as well as the Appendix discussing dialup networking.

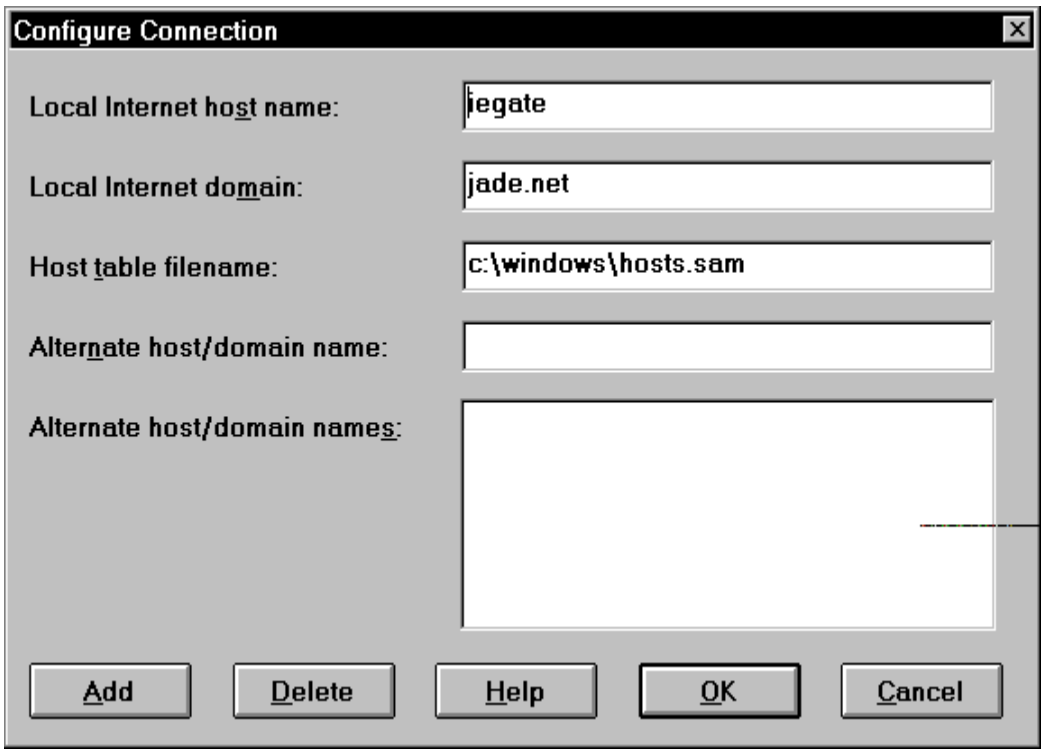
Send keep alive packets

For TCP connections that are made over a dialup connection (typically PPP or some ISDN connections) some stacks can be configured to timeout and automatically disconnect after a predetermined period with no network activity. Under these conditions, it is necessary for the gateway to keep the stack active if *SMTPD* is to continue to be able to receive incoming mail. If the *KeepAlive* option is enabled, *SMTPD* will send keepalive packets (a single UDP packet) to the discard port (9) of a remote host. The gateway will first look for a DNS server, followed by a sequential search for

any host other than the gateway itself in the hosts file to send the keepalive packets to. The keepalives are sent one packet approximately every 10 seconds.

CONFIGURE CONNECTION

This screen allows for the entry of connection and alternate host/domain name information for **Internet Exchange**:



Local Internet Hostname

This field records the Internet hostname of the gateway machine. In the above example, if the FQDN for the gateway machine is *iegate.jade.net*, the Local Internet Hostname would be *iegate.*

Local Internet Domain

This field records the Internet domain of the gateway machine. In the above example, if the FQDN for the gateway is *iegate.jade.net*, the Local Internet Domain would be *jade.net*.

Host Table Filename

This field records the location of the Internet host table for address resolution. Even if the DNS is used for name resolution, it is necessary that a host table be configured that contains at least the name and address for the gateway machine as well as for the default mail relay host. This will allow **Internet Exchange** to send mail to the default mail relay host for further routing in the event problems communicating with the name server(s) occur.

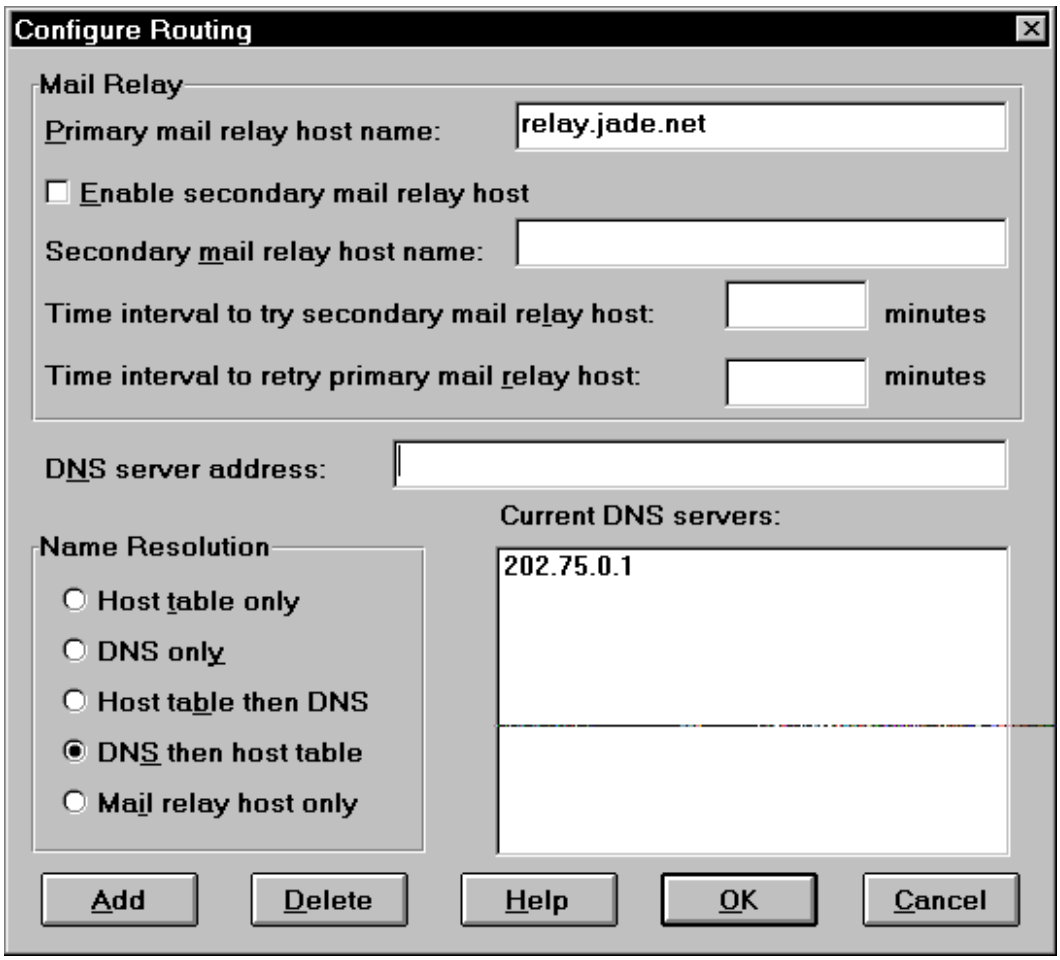
The default configured into **Internet Exchange** is *c:\ieccmail\hosts*. However, for Windows 95, this value should be set to *c:\windows\hosts* while for Windows NT, it should be set to *c:\windows\system32\drivers\etc\hosts*, if using the default locations.

Alternate Host/Domain Name

Sometime it is desirable for a gateway machine to be known by more than one fully qualified domain name (FQDN). This field is used to add additional names by which the gateway host is known. This list of alternate names contains entries that can be added to or deleted by using the appropriate buttons. All messages addressed to users at hosts identified in this list will be considered local when received by the gateway and will be sent to cc:Mail.

CONFIGURE ROUTING

This screen allows entry of routing information for **Internet Exchange**:



Primary mail relay host name

If *SMTPC* is unable to resolve a hostname by either DNS or host table lookup, it will route the message to this host for forwarding. This option is also used if routing is configured to mail relay host only.

Enable secondary mail relay host

If this checkbox is set, a secondary mail relay host can be configured for use when the primary mail relay host is unavailable.

Secondary mail relay host name

If primary mail relay host is down, then the gateway will route the message to this host instead.

Time interval to try secondary mail relay host

Number of minutes of unavailability after which the primary mail relay is considered offline and the secondary, if enabled, will be tried.

Time interval to retry primary mail relay host

The number of minutes before the gateway will try to revert to the primary mail relay host after it has been unavailable.

DNS Server Address

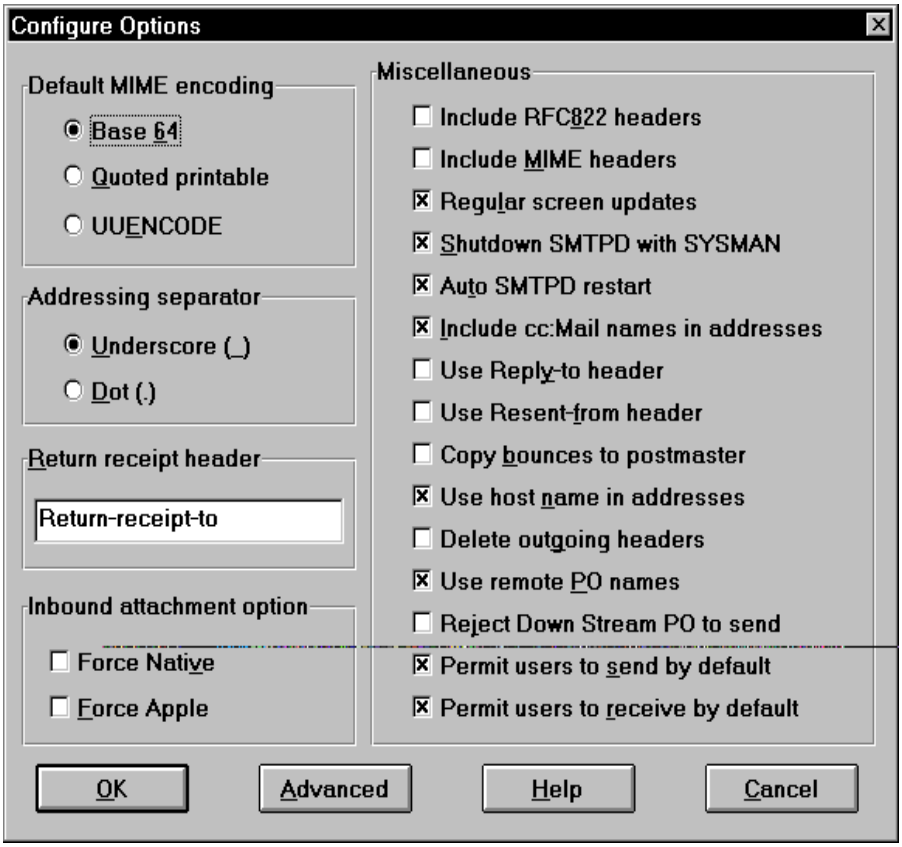
SMTPC will try contacting the list of configured DNS servers. Each address must be of the form a.b.c.d, where each number is between 0 and 255. (See Appendix B of the **Internet Exchange** for cc:Mail *Gateway Administrator's Manual, Version 2.0* for a discussion of IP addressing.)

Name Resolution

Any combination of DNS or host table lookup can be used, in any order. Where mail relay host only routing is not used, it is recommended that DNS be used if at all possible, as this will usually result in the most reliable routing and greatest throughput.

CONFIGURE OPTIONS

This screen allows entry of various options for **Internet Exchange**. The following screen shows the default values:



Default MIME Encoding

When encoding cc:Mail messages, *CCOUT* uses the MIME encoding information configured into **Internet Exchange** (see the next section for details). When a non Macintosh file with an unknown extension is encountered, it will be encoded using the default binary encoding. Choose the appropriate radio button based upon the capabilities of those sites which you communicate with most. Base64 is preferable for communicating with MIME-capable sites. Uuencode should be specified as the default if you intend on communicating with many sites that are not MIME-compliant. However, as uuencode/uudecode are not part of the MIME specification their widespread use is discouraged.

Addressing Separator

When constructing Internet addresses for cc:Mail users without an explicit entry in the *SMTP.ADR* file, all spaces are usually converted to underscores as spaces are not valid in Internet mail addresses. Some sites prefer to use dots instead of underscores - this option allows for choosing between the two.

Return Receipt Header

This parameter allows the value of the Internet Return Receipt header to be specified. Using the default value of *Return-Receipt-To:* allows compatibility with the UNIX *sendmail* program and the Lotus SMTPLINK product. However, there are problems involved with this approach. S*endmail* uses the header to request notification of message delivery at the transport level, while SMTPLINK uses it to signify that the message has been opened (and possibly read) by the recipient.

Choosing a different value will sidestep this problem, but will also ensure that the return receipt function is portable only between IMA gateways that settle upon the same value to use.

Force Native

By checking this option, inbound Macintosh attachments (in BinHex, MacMIME or uuencoded AppleSingle format) are stripped of their header and (if present) resource fork before being attached to messages in the cc:Mail Post Office. If this is not done, some applications (like Excel 4 for Windows) may refuse to open the resulting file. See Chapter 3 for a more detailed discussion of this issue.

Force Apple

By checking this option, inbound non-Macintosh attachments are given a dummy header and converted into AppleSingle cc:Mail attachments before being attached to messages in the cc:Mail Post Office. The type and creator are obtained from the MIME table prepared with the *Configure MIME* dialogue box. See Chapter 3 for a more detailed discussion of this issue.

Include RFC822 Headers

In normal operation, **Internet Exchange** discards RFC822 headers after the messages have been imported into cc:Mail. This option allows all such headers to be retained in the message as a separate attachment.

Include MIME Headers

In normal operation, **Internet Exchange** discards MIME bodypart headers after they have been processed. This option allows all such headers to be retained in the message as separate attachments.

Regular Screen Updates

Screen updates occur at regular intervals during normal gateway operations. If one or more of the mail queues grows very large, *Regular Screen Updates* should be turned *off*. Otherwise the system will spend too much time re-reading and re-displaying the queues.

Shutdown SMTPD With SYSMAN

When set, this will shut down the SMTP daemon when the *SYSMAN* program exits.

Auto SMTPD Restart

This allows *SYSMAN* to automatically restart the SMTP daemon if and when it should exit. The default is *on*.

Include cc:Mail names in Addresses

Turning this option off removes the cc:Mail user name from Address field leaving only the user's Internet address. The default is *on*.

Use Reply-To Header

This option makes use of the *Reply-to:* field by copying it to the *From:* field on all incoming mail. Otherwise this information is lost, as cc:Mail has no concept of a *Reply-to:* field. Use of this option will result in the loss of the original *From* field, if this field is different from the *Reply-to* field.

Use Resent-from header

In the versions of **Internet Exchange** prior to 2.0, if a *Resent-From:* header was present on the incoming message, it was used for the cc:Mail *From* field instead of the *From:* header. This option allows this behavior to be turned on or off as needed. Use of this option will result in the loss of the original *From* field, if this field is different from the *Resent-from* field.

Copy Bounces To Postmaster

Setting this option directs all bounced messages to the local postmaster, as well as the original sender of the message.

Use Hostname In Address

This option determines whether the local hostname is included in Internet addresses for cc:Mail users. As an example, the user John Smith might appear to the outside world as *John_Smith@iegate.jade.net* with this option set, and as *John_Smith@jade.net* with the option not set.

Delete Outgoing Headers

When a MIME message is imported into cc:Mail and either RFC822 and/or MIME headers are included, extra text items are created containing these headers. When such messages are resent out to the Internet, these text items are not useful, and often confuse the recipient. Setting this option automatically deletes these header attachments from outgoing messages.

Use Remote PO Names

Using default address mapping, a message from a user at a remote Post Office of "Sales" will appear to be sent from a user in the following form:

John_Smith_at_Sales@iegate.jade.net

This is not attractive. By disabling this parameter, the message will appear to come from:

John_Smith@iegate.jade.net

which is much tidier. To ensure that replies to this message will be returned to the sender, there must be an entry in the gateway Post Office for the user. This can most easily be accomplished by using Lotus ADE (Automatic Directory Exchange).

Reject Down Stream PO to send

Setting this option will disable users from downstream post offices to send messages to the Internet. Thus, only users from the local post office can send messages to Internet.

Permit users to send by default

If send permission is not set for a user in the alias database and directory database, this option determines whether the user can send messages to the Internet.

> **NOTE: Internet Exchange Workgroup Edition** does not allow for the configuration of this parameter, and the default is NO.

Permit users to receive by default

If receive permission is not set for a user in the alias database and directory database, this option determines whether the user can receive messages from the Internet.

> **NOTE: Internet Exchange Workgroup Edition** does not allow for the configuration of this parameter, and the default is NO.

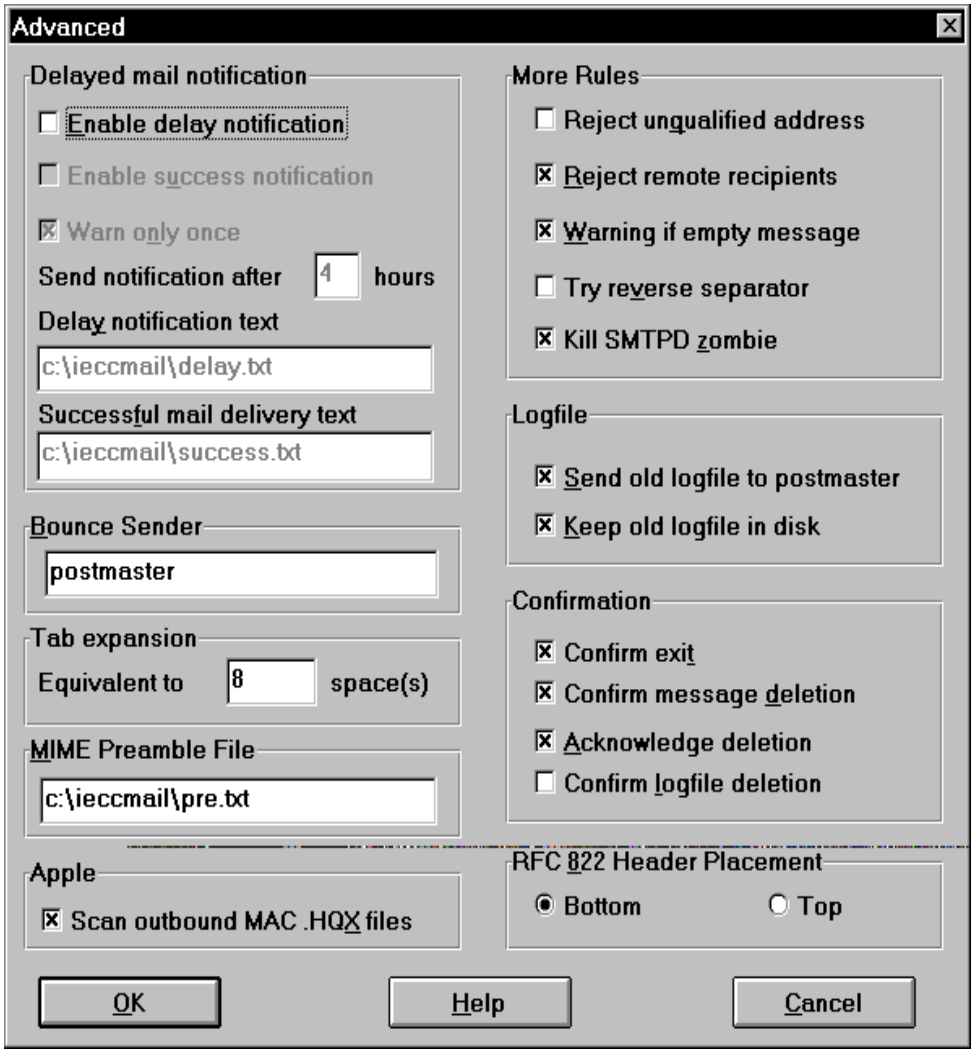
IMPORTANT: When the default permission to receive is set to "NO", Internet addresses can only be converted into a corresponding cc:Mail user name if a corresponding entry is present in either the Alias or the Rules-Based databases. This implies that some cc:Mail users mentioned as recipients (either "To:", or "Cc:") on outgoing messages may not have any associated valid Internet address. By default, they are not considered for conversion while building the recipient headers of the resulting Internet message. In order to enable **Internet Exchange** to generate references to these users, the following variable should be changed in the *IMA.INI* file:

```
[Options]
IncludeNonRepliableAddresses=YES
```

By setting this option to YES, the default mapping will be used anyway for giving visibility over the original list of recipients. Any attempt to reply from the Internet side however without editing out the unrepliable addresses will result in bounces (without affecting the delivery to the repliable ones).

Advanced Options

A set of advanced options can be accessed through the *Advanced* button:



The *Delayed mail notification* section allows the setting of many options related to handling messages that cannot be delivered on the first attempt:

Enable delay notification

When this option is set, **Internet Exchange** will warn the sender when a message cannot be delivered for a long period of time. This time is defined by the *Send notification after ... hours* option described below.

Enable success notification

If this option is set, **Internet Exchange** will notify the sender when a delayed message has been successfully sent.

Warn only once

If this option is set, the gateway does not send additional notifications for repeated delays of the same message. Otherwise, delay notifications will be sent regularly until the message is successfully delivered.

Send notification after ... hours

Defines the threshold of what should be considered delayed delivery.

Delay notification text

The pathname of the file containing the message to be used to notify of a delayed message delivery. If no file name is specified or no file is found at the specified path, the following default warning message text will be used:

> The Internet Exchange gateway (localhost.domain) has come across problems delivering to the following Internet recipient(s): recip1, recip2. This mail message has been delayed for *n* hours. The gateway will continue to retry the message and may send it back if it cannot be eventually delivered.

Successful mail delivery text

The pathname of the file containing the message that will be sent to postmaster when the gateway eventually delivers a delayed message. If none is specified or if no file is found at that path, the following default message text will be sent:

> The Internet Exchange gateway (localhost.domain) has successfully delivered your delayed message to the Internet recipient(s): recip1, recip2, ...

Bounce Sender

The sender of the messages generated by the gateway to report undeliverable mail. The default is postmaster which is highly recommended.

Tab expansion

The number of spaces used to replace tab characters in incoming text messages. If set to zero, tabs will not be replaced. As some cc:Mail clients have trouble displaying tab characters, this option allows them to be replaced by spaces.

MIME Preamble File

MIME multipart messages contain an initial section known as the preamble, where a short optional text useful to non-MIME gateways and user agents can be stored. This section resides between the RFC822 headers and the first MIME body part. If this option identifies an existing file, the contents of this file will be used as the preamble in outgoing messages. If set to a nonexistent file, no preamble is used. If not set, no preamble is used.

Scan outbound MACINTOSH .HQX files

In outgoing messages, check whether the message body contains files in BinHex format and, if found, use the information in the header to prepare the proper MIME headers.

Reject Unqualified Addresses

SMTPD will check recipient and sender addresses for a proper domain part, refusing to receive messages where it is absent. e.g. *user@host.com* is accepted but *user* is rejected. This option can be useful in encouraging users to always use FQDNs when sending email to the Internet.

Reject Remote Recipients

SMTPD will reject incoming messages for remote Internet recipients. This is to prevent remote sites from trying to spoof messages by rerouting them through the gateway back out to the Internet.

Warning if empty message

This will cause empty outbound messages to trigger a warning to the local postmaster. The warning text is:

> *Warning: your message went out the cc:Mail gateway with an empty message body.*
>
> > *If you intentionally sent an empty message, disregard this warning.*
> > *If you included a reply in an old header body part, it was purged.*
>
> Key headers from the message that was sent follow.

Try reverse separator

This will cause both address separators (dot / underscore) to be tried with incoming addresses during default address translation. This can be useful if the local site changes its preferred separator and still wishes addresses with the old separator to be valid.

> **NOTE:** This option is not available in the *Workgroup Edition.*

Kill SMTPD zombie

When this option is set, *SMTPD* checks the value of the SMTPDmainSocket option in the config section of IMA.INI upon startup. If this is *not* set to NONE, the number indicates the main socket used by *SMTPD* when it shutdown prematurely last time around. An attempt to close this socket is performed, so that *SMTPD* does not get an *address already in use* error when restarted.

Send old logfile to postmaster

If this option is checked, all old logfiles are automatically mailed to the postmaster.

Keep old logfile on disk

Setting this option prevents the deletion of old log files.

Confirm exit

Setting this option results in the gateway asking for confirmation before a manual shutdown.

Confirm message deletion

Setting this option results in the gateway asking for confirmation before deleting a message.

Acknowledge deletion

Setting this option results in the gateway presenting acknowledgment after successfully deleting a message.

Confirm logfile deletion

Setting this option results in the gateway asking for confirmation before deleting the log file.

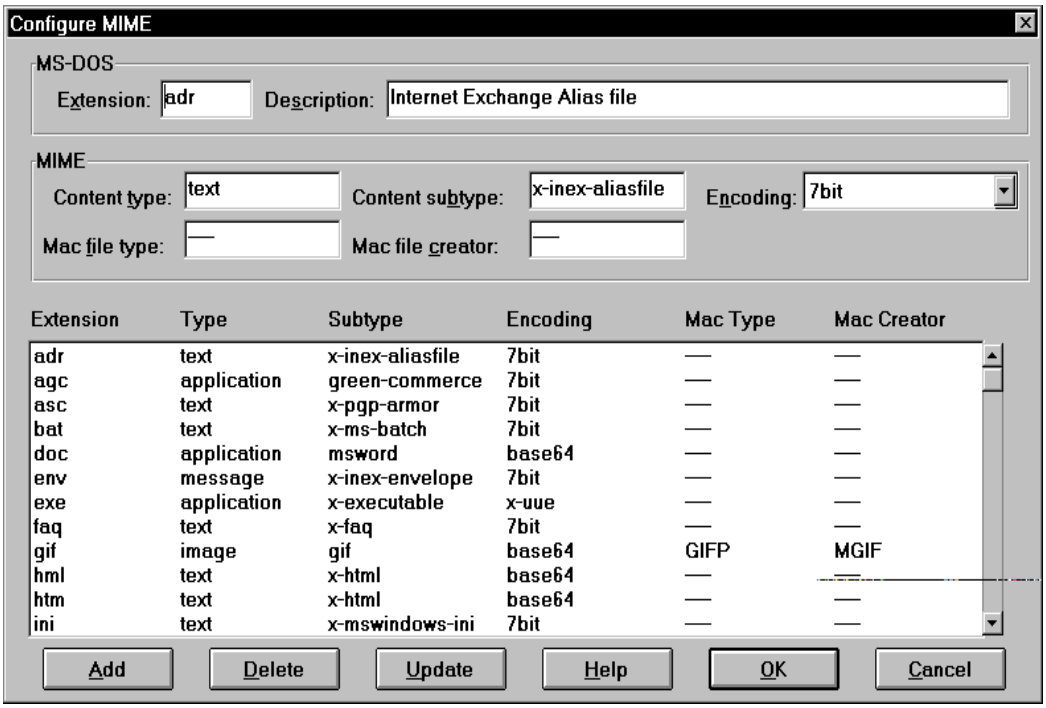
RFC 822 Header Placement

Controls where the RFC822 headers will be attached from an incoming message. i.e. before the first attachment or after the last.

CONFIGURE MIME

When cc:Mail users send messages containing attachments to recipients on the Internet, it is necessary to encode the message and attachments according to the MIME standard. The MIME standard provides a framework for both the encapsulation of attachments within a single message, as well as the encoding of these attachments.

Internet Exchange gives the gateway administrator full control over how file attachments are encoded for messages originating within cc:Mail. An internal table is maintained by the gateway that provides for the mapping between DOS file extensions and MIME content type/subtype and encoding methods. Information is also maintained for communicating with Macintoshes.



The *Configure MIME* screen gives the administrator the ability to modify the manner in which the gateway handles specific file types and to extend its abilities by adding new file types. **Internet Exchange** ships with the standard set of MIME types and subtypes as defined by the Internet Assigned Numbers Authority (IANA) pursuant to RFC1590 ("Media Type Registration Procedure"). This set, which is periodically updated, is available at:

ftp://ftp.isi.edu/in-notes/iana/assignments/media-types/media-types

To enter new MIME mapping rules, the gateway administrator simply fills in the fields *Extension*, *Description*, *Content type*, *Content subtype*, *Encoding*, *Macintosh file type* and *Macintosh file creator* and saves the entry by pushing the *Add* button. Existing entries can be edited by double clicking, and then modifying the appropriate fields. As with a new entry, the modified entry is entered by pushing the *Add* button. Mapping rules can be removed by selecting the appropriate rule and then pushing the *Delete* button. For more information on how to construct MIME content types/subtypes, please refer to Appendix A of the **Internet Exchange** *Gateway Administrator's Manual, Version 2.0*.

Using the above information, an outgoing GIF attachment will result in the following MIME header:

Content-Type: image / gif; name="world.gif"

In the case of an outgoing GIF sent from a Macintosh, the result might appear as follows:

Content-Type: image / gif; name="world.gif"; type="GIFP:MGIF"

This presumes that the Macintosh file type and creator are *GIFP* and *MGIF* respectively.

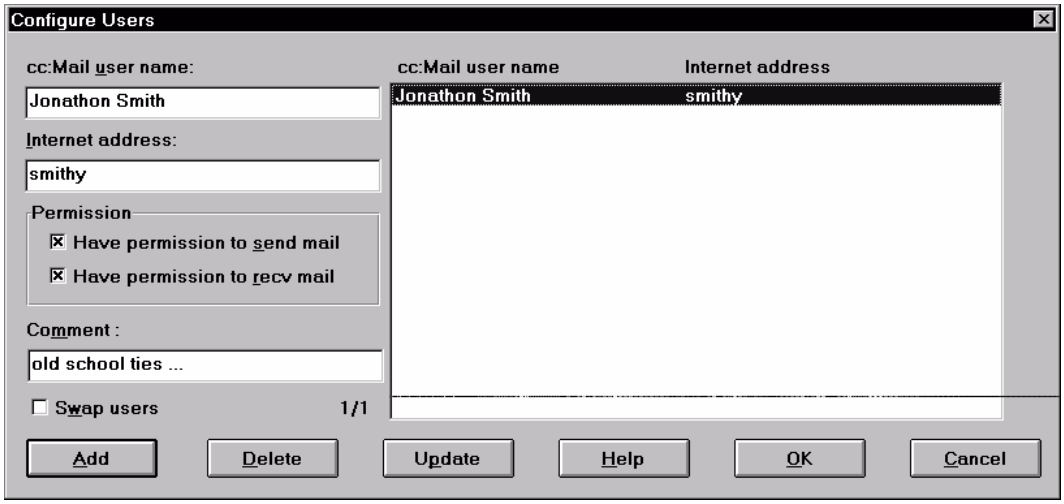
When the gateway comes across an attachment type that is not in the MIME mapping table, it applies the default binary encoding method specified in the *Configure Options* screen. The resulting MIME header would look similar to:

Content-Type: application / octet-stream; name="test.abc"

CONFIGURE USERS

This screen allows for the configuration of cc:Mail username to Internet address mapping on a per user basis. In the absence of an entry in the *SMTP.ADR* file, **Internet Exchange** will apply the addressing rules. If still unsuccessful, it will apply the default address mapping (*Enterprise Edition* only), which converts the cc:Mail username to:

firstname_lastname@gateway.domain



When messages are received through SMTP for delivery into cc:Mail, **Internet Exchange** first consults the User Mappings Alias Database for per user address translation rules. If an entry is found, the corresponding cc:Mail address is used for submission into cc:Mail. If an entry is not found, rule based addressing is tried, then the default address translation rule described above is used if that is unsuccessful.

When messages are sent from cc:Mail to the Internet, **Internet Exchange** will consult the User Mappings Alias Database to match possible *To:*, *From:*, and *Cc:* addresses. If matches are found, these addresses are replaced with the corresponding Internet address prior to submission to the Internet. If an entry is not found, rule based addressing is tried. If unsuccessful, the default address translation rule is applied in order to construct a valid Internet address.

To add a new entry, enter the cc:Mail username, the Internet address, and an optional comment. By checking the send and receive permissions checkboxes, local users can be given different capabilities.

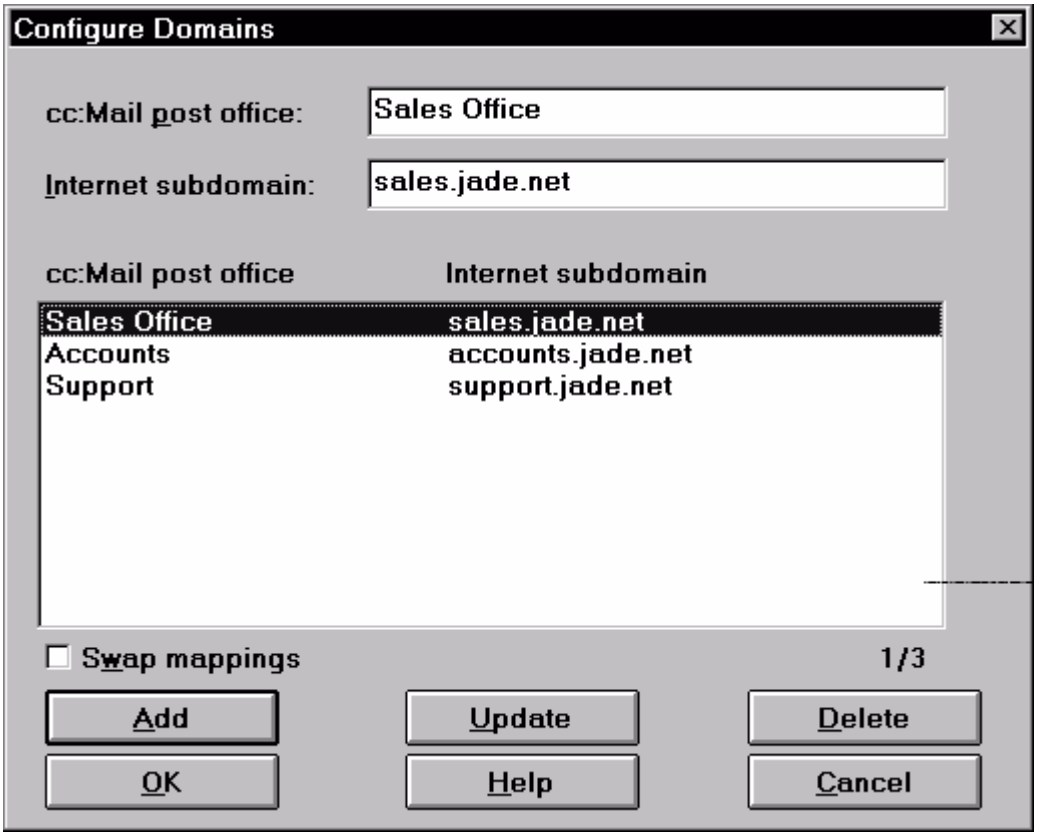
NOTE: Internet Exchange *Workgroup Edition* supports a maximum of 100 registered users of the gateway. This number is determined by adding the number of unique cc:Mail users found in the Alias and Rules Based Addressing databases. The gateway will check for the limits when either the User Alias or Rules Based Addressing databases are modified. If the number of users exceeds the maximum allowed, the respective database update will not be made. Manual regeneration of either database which results in an error will result in a dialog box to the administrator indicating the problem. Errors detected during Dynamic Conversion will result in the error message being mailed to the gateway administrator.

Swap users

When this option is checked, the position of user mappings can be changed. This is done by highlighting a specific entry, and using Up arrow, Down arrow, Home, End, Page up and Page Down. These keys will move the selected entry in that direction. The first entry for a particular cc:Mail user will have the highest priority and will be used for outgoing address translation.

CONFIGURE DOMAINS

This screen allows the creation of Internet-style subdomains within the local cc:Mail environment. These subdomains are mapped to cc:Mail Post Offices (connected through Router to the main PO) which are hidden from the Internet by the gateway. This style of post office to Internet subdomain name mapping is useful when remote post offices are communicating with the gateway, and the cc:Mail post office routing information needs to be maintained across the gateway.



In the following example, three cc:Mail post offices are hidden behind the main post office (*Jade Networks*). The hidden cc:Mail post office names are *Sales Office*, *Accounts*, and *Support.*

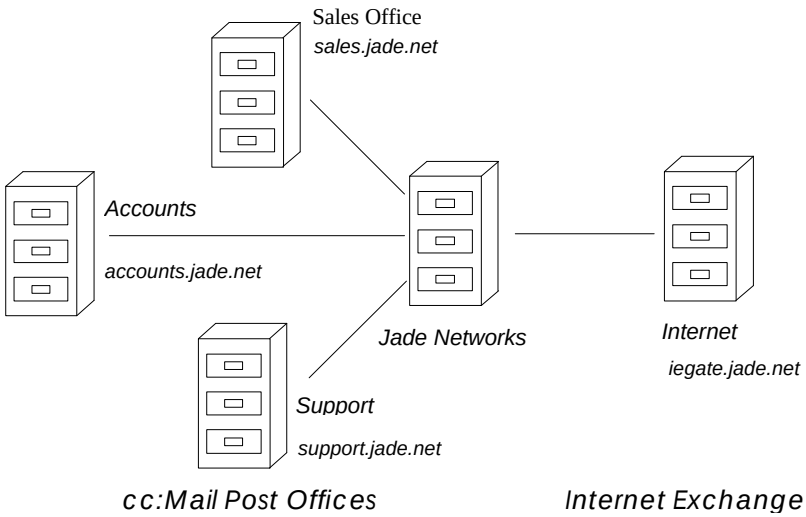


Figure 2.1: cc:Mail Subdomain Name Mapping

While the cc:Mail post office names are descriptive and legal in the cc:Mail domain, they do not make for good domain name components on the Internet, due to the frequent use of spaces. Since the fully qualified domain name of the **Internet Exchange** gateway machine is *iegate.jade.net*, the remote post offices will appear as subdomain within this domain.

The *Configure Domains* screen gives the gateway administrator the ability to perform

these post office to subdomain mappings. When configured as above, cc:Mail messages that originate in the *Sales Office* post office will appear to have come from *user@sales.jade.net* when they reach the Internet. Internet messages delivered to *iegate.jade.net* with a recipient address on *sales.jade.net* will be delivered to the post office *Jade Networks* for further routing within the cc:Mail domain to the post office *Sales Office.*

Swap mappings

When this option is checked, the position of subdomain mappings can be changed. This is done by highlighting a specific entry, and using Up arrow, Down arrow, Home, End, Page up and Page Down. These keys will move the selected entry in that direction.

CHAPTER 8
RULES BASED ADDRESSING

INTRODUCTION

The use of rules based addressing allows increased flexibility with respect to what kinds of Internet addresses are used with **Internet Exchange**. While a very simple method of address formatting is provided by way of the default separator, the use of rules based addressing is much more powerful. Rules can be setup so that a number of incoming address formats are accepted, allowing many different addresses to be delivered correctly. In the same spirit, a preferred format can be selected for translating outgoing cc:Mail addresses into Internet addresses.

HOW RULES BASED ADDRESSING WORKS

An addressing rule specifies how to translate between a cc:Mail username and the corresponding Internet address. It is composed of different combinations of the first, middle and last names, either partial or in full, as well as an optional separator of an underscore (_) or a dot (.).

Here are some examples of addressing rules, with the resulting address corresponding to a cc:Mail username of Jonathan Andrew Smith:

Address Rule	***Example Name***
FA_M1_LA	Jonathan_A_Smith
F1M1LA	JASmith
F1_MA_LA	J_Andrew_Smith
F1.LA	J.Smith

In the first example above, if an Internet message comes in addressed to Jonathan_A_Smith@jade.net, then it will be delivered to the local user Jonathan Andrew Smith. If there are many addressing rules specified, then each format of incoming address will be recognized. i.e. any of the addresses on the above right will be translated into Jonathan Andrew Smith, assuming there are no name collisions (see later).

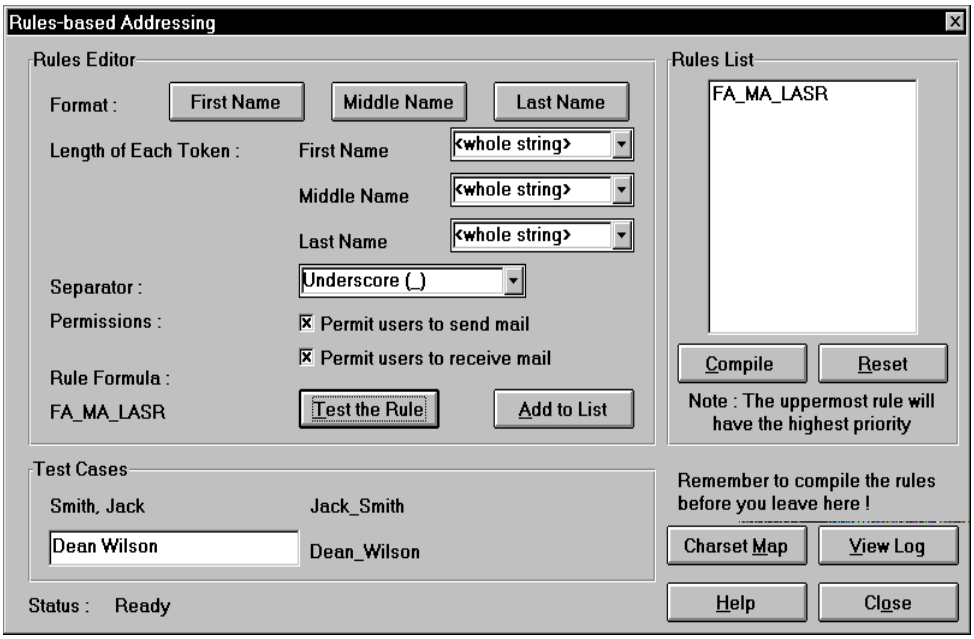
It is not necessary to understand the abbreviated format on the above left, as the creation of addressing rules is done by way of a detailed dialog box as explained in the next section.

At the end of each rule are optional characters, indicating whether the cc:Mail user has send and/or receive permission. The letters present represent the following permissions:

Permissions	***Send***	***Receive***
Allowed	S	R
Banned	X	X

Once the addressing rules have been established, they must be compiled to produce an address directory database. This greatly increases the speed with which **Internet Exchange** can process messages. If any changes are made to either the cc:Mail directory or the addressing rules, a new compilation *must* be run by the administrator. **Internet Exchange** will not use the changes until a new compilation occurs.

CONFIGURING RULES BASED ADDRESSING



The Address Rules configuration dialog box allows easy maintenance of the rules based address system. The first step is to decide upon the preferred format of outgoing addresses. This will look similar to the examples given above. Each rule can be built from up to three separate name parts, optionally separated by either a dot(.) or an underscore(_). The three name parts are FirstName, LastName, and MiddleName(s). If there are more than three parts to the name, then each extra name is added to the MiddleNames(s) part. Any name in the standard cc:Mail format of *LastName,FirstName* will initially be reversed to appear as *FirstName LastName.*

Each of the three buttons on the *format* line can be cycled through the available name parts: first name, middle names, last name, or not used. However, these can only be used in the rule once. e.g. you cannot include the first name twice in a single rule.

After selecting which parts of the cc:Mail username to use, the length of these name parts can be chosen by the set of combo boxes below. Each name part can be either the full name, or any number of initial characters up to nine.

Next, the separator can be chosen, either the underscore(_), the dot(.) or no separator.

Next, the send / receive permissions can be specified using the appropriate checkboxes.

At any time, the current rule can be tested by pushing the *test* button. This shows the results of applying the current rule on two test cases at the bottom of the screen. One of the test cases is fixed, while the other can be changed.

Once the addressing rule is complete, it can be added to the rules list by pushing the *Add to List* button. The new rule will be appended to the list of current rules in the listbox on the righthand side of the screen. The *reset* button can be pushed at any time to delete all the current addressing rules.

Note: the first rule in the list will be used to generate outgoing Internet addresses.

After all new rules have been added, the compile button must be pushed to compile the addressing rules into the address directory database. If this operation is not done, any new addressing rule will be ignored. A logfile is kept showing the results of the compilation process, and this maybe inspected by pushing the *View Log* button.

NOTE: Internet Exchange *Workgroup Edition* supports a maximum of 100 registered users of the gateway. This number is determined by adding the number of unique cc:Mail users found in the Alias and Rules Based Addressing databases. The gateway will check for the limits when either the User Alias or Rules Based Addressing databases are modified. If the number of users exceeds the maximum allowed, the respective database update will not be made. Manual regeneration of either database which results in an error will result in a dialog box to the administrator indicating the problem. Errors detected during Dynamic Conversion will result in the error message being mailed to the gateway administrator.

Name Collisions

In some cases, rule compilation produces a clash in names. There are several situations in which this can occur:

for example:	rules:	F1M1LA F1M2LA
	user:	Barry Dilmann
	mappings:	BDilmann (twice)

In this case, the duplicate mapping is discarded, leaving a single mapping. If two rules produce the same mapping with two separate users, the mapping with the highest priority will used.

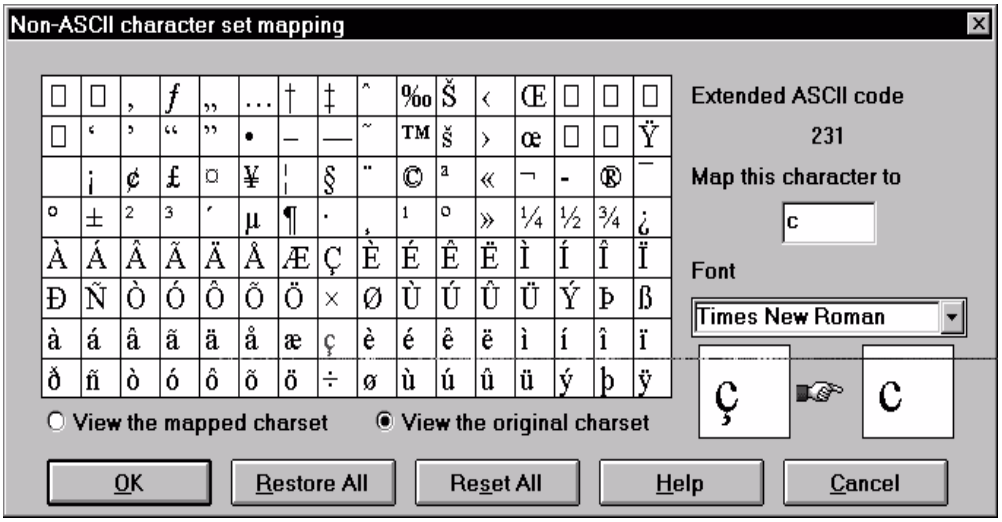
Sometimes a single rule will produce a clash for two users:

for example:	rule:	F1LA
	users:	John Quentin Thompson John Martin Thompson.
	mapping:	JThompson

Here, the mapping is discarded completely, so it would be necessary to add entries in SMTP.ADR so that each user has a unique mapping.

CHARACTER SET MAPPING

Extended ASCII characters (codes 128 - 255) are not allowed within headers of Internet messages. **Internet Exchange** allows these to be converted into standard ASCII characters (codes 0 - 127) by selecting the *Charset Map* dialog box. The administrator can build a customized mapping to convert any Extended ASCII characters into standard ASCII characters to be used in outgoing message headers. This can be especially useful when using international character sets within cc:Mail.



The main character set display in the dialog can display either extended ASCII, or the set of characters that it is mapped to. This can be changed by selecting the appropriate radio button underneath the character set display.

An extended ASCII character can be mapped into at most two standard ASCII characters. A new mapping may be added by selecting an extended ASCII character in the character set display and typing a corresponding standard ASCII character into the *map* text box. Once this has been done, the corresponding character is displayed in the box after the small hand.

The *font* button allows a different character set to be displayed. This can be useful when working with international character sets.

There are two buttons that allow recent changes to be discarded. *Restore All* returns settings to how they were before this editing session, while *Reset All* removes all character mappings.

Note: character set mapping can only be configured if at least one addressing rule has been defined. Once defined, it is valid across all rules.

CHAPTER 9
PEER DOMAIN CONFIGURATION

INTRODUCTION

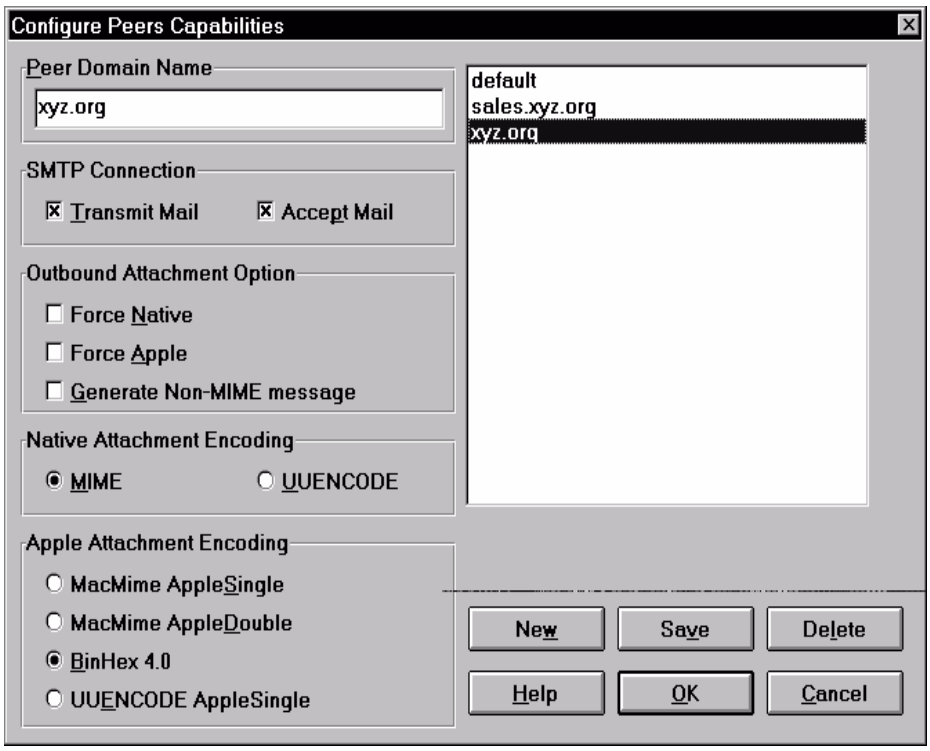
The Internet hosts with which **Internet Exchange** communicates will often have different capabilities, such as which email formats they can accept. The *Configure Peer* dialog box allows such information to be recorded and used in preparing outgoing messages for the Internet. This will ensure that messages sent to the Internet are able to be successfully decoded by their recipients.

This information is stored as a list of peers for which certain capabilities apply. These capabilities apply to a specific domain and all its subdomains, unless a more specific capability exists within the database. For example, take the following capabilities:

Peer	***Capability***
xyz.org	BinHex 4.0
sales.xyz.org	MacMime AppleSingle

For outgoing messages with Macintosh attachments, any message sent to xyz.org will be encoded with BinHex 4.0. Messages going to test.xyz.org will also be encoded with BinHex 4.0. However, messages going to sales.xyz.org will be encoded using MacMime AppleSingle , as will messages to m1.sales.xyz.org.

CONFIGURING PEER CAPABILITIES



The initial text box allows entry of the required domain. On the right hand side, there is a listbox showing all domains for which a peer capability has been defined. By selecting an existing peer from this listbox, all the capabilities for it will be displayed in the dialog box. These capabilities can be divided into the following groups.

SMTP connection

A checkbox is provided to indicate whether Internet Exchange is allowed to send to and/or receive messages from this domain. If **Internet Exchange** is not allowed to transmit mail to a remote site, *CCOUT* will bounce any message destined for that host. If **Internet Exchange** is not allowed to receive mail from a remote host, *SMTPD* will reject a HELO command from that host with the following response:

550 host sales.xyz.org is not authorised to connect to iegate.jade.net

Outbound Attachment Option

Three checkboxes are offered, specifying how to encode general attachments in outgoing messages. These are:

Force Native

This option will result in all Apple attachments being stripped of their headers and resource fork. This will allow non Macintosh sites to access the information easily.

Force Apple

This options will result in all non Apple attachments being changed to Apple format. This involves adding a header and an empty resource fork, and then encoding using the Apple encoding specified below. This option might be useful when **Internet Exchange** is talking primarily to a network of Macintoshes.

Generate non MIME message

This option ensures that no MIME messages are generated for this peer. This can be useful when communicating with older email systems that do not understand MIME. In this case, either UUENCODE or BinHex 4.0 is used to encode binary attachments.

Native Attachment Encoding

MIME

This option specifies that non-Apple attachments are encoded using the MIME standard.

UUENCODE

This option specifies that non Apple attachments are encoded using the older UUENCODE format. The *Generate non MIME message* determines whether MIME headers will be generated for the message.

Apple Attachment Encoding

MacMime AppleSingle

This option specifies that outgoing Macintosh attachments are encoded using the MacMime AppleSingle standard.

MacMime AppleDouble

This option specifies that outgoing Macintosh attachments are encoded using the MacMime AppleDouble standard.

BinHex 4.0

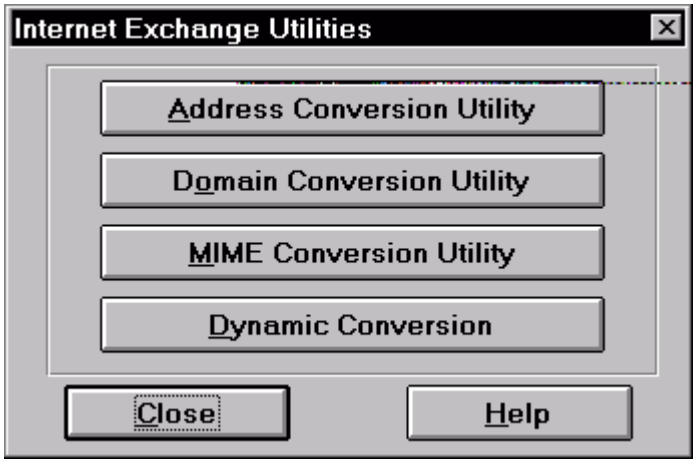
This option specifies that outgoing Macintosh attachments are encoded using the BinHex 4.0 standard. The *Generate non MIME message* determines whether MIME headers will be generated for the message.

UUENCODE AppleSingle

This option specifies that outgoing Macintosh attachments are encoded using the AppleSingle standard, using UUENCODE instead of MacMime. The *Generate non MIME message* determines whether MIME headers will be generated for the message.

CHAPTER 10
UTILITIES

With the exception of the *Message Database Rebuild* and *Message Conversion* utilities, all utilities are accessible via both the Utilities button on the main screen as well as through the program manager. The two referenced utilities above are available only through the program manager and can be found in the *Internet Exchange* program group. The Utilities button on the main screen allows access to various programs that help with maintaining **Internet Exchange**:

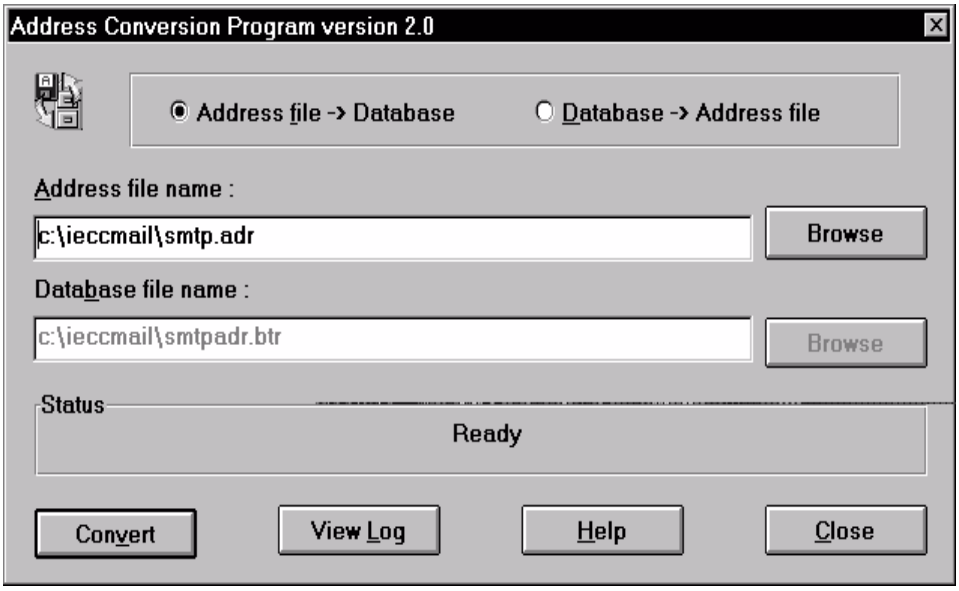


ADDRESS CONVERSION UTILITY

The *CONVADR* program will convert between the text *SMTP.ADR* file and the new User Mapping Database, introduced in **Internet Exchange** version 2.0. It is most useful when an external program has been written to create a new version of *SMTP.ADR* regularly. Once generated, this new version of *SMTP.ADR* needs to be converted to the User Mapping Database, using this tool. An alternative to running this manually is to enable an option within the *Dynamic Conversion* section of the *Help* screen. This will enable *SYSMAN* to regularly check for changes in *SMTP.ADR* and run this conversion program automatically if any changes are detected. Setting this option greatly reduces the workload associated with maintaining the User Mapping Database .

Another option within the *Dynamic Conversion* section of the *Help* screen controls whether to create *SMTP.ADR* in the old or new format. The new format, introduced in **Internet Exchange** version 2.0, adds send and receive permissions as well as an

optional comment field to each record. Converting to the old format can be useful for interoperability with **Internet Exchange** versions prior to 2.0.



The two radio buttons at the top of the screen indicate which direction the conversion will run:

Address file -> Database

Converts the old *SMTP.ADR* user mapping file into the new indexed Directory Database, deleting any current information.

Database -> Address file

Saves the information from the User Mapping Database to an ASCII file in the same format as *SMTP.ADR*. The destination file is overwritten. Either the alias database (see *Configure Users*) or the rules based directory database (see *Configure Address Rules*) may be converted. An uptodate *SMTP.ADR* file can serve as a backup in case the User Mapping Database ever becomes corrupted.

Convert

Perform the conversion.

View Log

Errors and related information during conversion are logged to the text file *SMTPADR.LOG* in the main directory.

Close

Close the utility.

SMTP.ADR Format

There are two formats that this utility can work with. The old format of *SMTP.ADR* was used in **Internet Exchange** versions 1.04b and earlier. However, for **Internet Exchange** version 2.0 the send and receive permissions and comment field have been added to the databases. This conversion utility handles both formats.

The old format of SMTP.ADR (for **Internet Exchange** version 1.04b or earlier) :

cc:Mail_Name<=>Internet_Name

e.g. John_Smith <=>J_Smith

The new format of SMTP.ADR (introduced with **Internet Exchange** version 2.0)

cc:Mail_Name<=>Internet_Name;<S/X><R/X>;<comment>

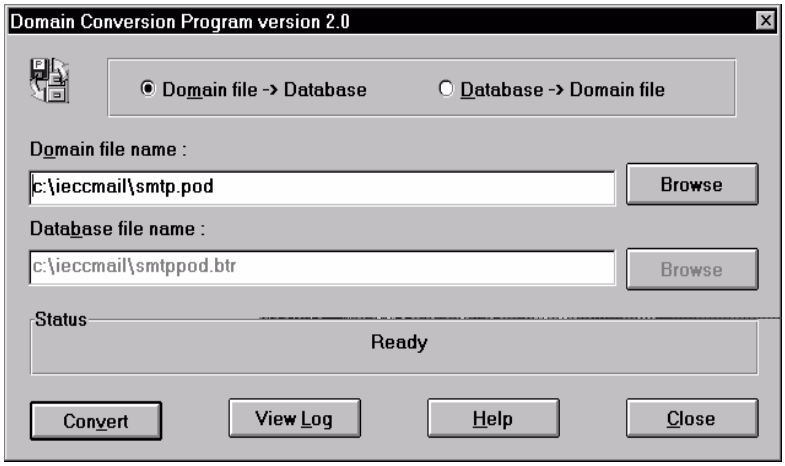
e.g. John_Smith<=>J_Smith;SR;Accountant
(John Smith is permitted to send and receive mail)

When an old format *SMTP.ADR* is converted to the new User Mapping Database, the send/receive permissions are enabled, and the comment field is left blank.

NOTE: **Internet Exchange** *Workgroup Edition* supports a maximum of 100 registered users of the gateway. This number is determined by adding the number of unique cc:Mail users found in the Alias and Rules Based Addressing databases. The gateway will check for the limits when either the User Alias or Rules Based Addressing databases are modified. If the number of users exceeds the maximum allowed, the respective database update will not be made. Manual regeneration of either database which results in an error will result in a dialog box to the administrator indicating the problem. Errors detected during Dynamic Conversion will result in the error message being mailed to the gateway administrator.

DOMAIN CONVERSION UTILITY

The *CONVPOD* program will convert between the old text *SMTP.POD* file and the new Domain Mapping Database, introduced with **Internet Exchange** version 2.0. It is most useful when an external program has been written to create a new version of *SMTP.POD* regularly. Once generated, this new version of *SMTP.POD* needs to be converted to the Domain Mapping Database, using this tool. An alternative to running this manually is to enable an option within the *Dynamic Conversion* section of the *Help* screen. This will enable *SYSMAN* to regularly check for changes in *SMTP. POD* and run this conversion program automatically if any changes are detected. Setting this option greatly reduces the workload associated with maintaining the Domain Mapping Database .



The two radio buttons at the top of the screen indicate which direction the conversion will run:

Domain file -> Database

Converts the *SMTP.POD* user mapping file into the new indexed Domain Mapping Database, deleting any current information.

Database -> Domain file

Saves the information from the Domain Mapping Databases to an ASCII file in the same format as *SMTP.POD*. The destination file is overwritten. An uptodate *SMTP.POD* file can serve as a backup in case the Domain Mapping Database ever becomes corrupted.

Convert

Perform the conversion.

View Log

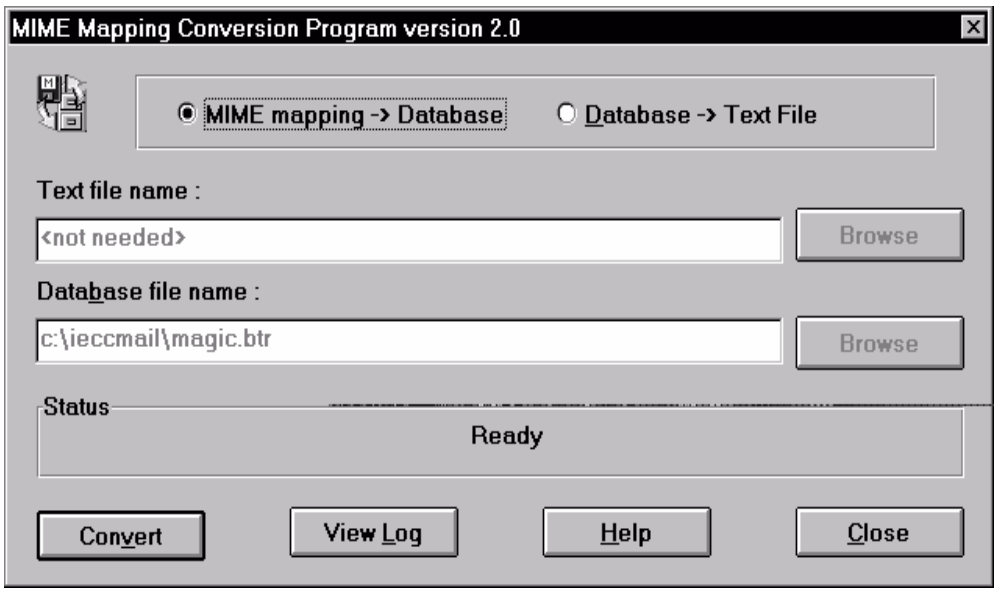
Errors and related information during conversion are logged to the text file *SMTPPOD.LOG* in the main directory.

Close

Close the utility.

MIME MAGIC MAPPING UTILITY

The *CONVMIME* program will convert between the text found in the Mime section of *IMA.INI* and the new MIME Mapping Database, introduced with **Internet Exchange** version 2.0. It can convert in either direction, and can produce an external text file containing the mappings as well.



MIME Mapping -> Database

Deletes any existing MIME Mapping Database and creates one based on the MIME mapping information found in the Magic section of the IMA.INI file. The new database fields for which there is no information are left empty.

Database -> Text File

Saves the information contained in the MIME Mapping Database to an ASCII file, suitable for inclusion in the Magic section of the IMA.INI file. An uptodate text file can serve as a backup in case the MIME Mapping Database ever becomes corrupted.

Convert

Performs the conversion.

View Log

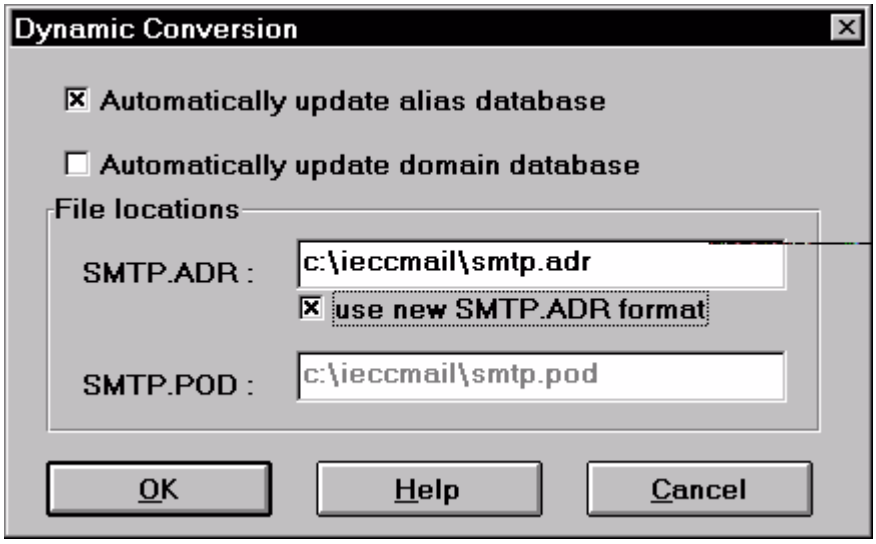
Errors and related information during conversion are logged to the text file *MAGIC.LOG* in the installation directory.

Close

Close the utility.

DYNAMIC CONVERSION UTILITY

The **Internet Exchange** System Manager (*SYSMAN)* program can automatically check the time stamps of the *SMTP.ADR* and *SMTP.POD* files, and if changes are detected, will run the conversion utilities automatically to rebuild the related databases. To change how often these checks are made, change the *system schedule* option in the *Configure Schedules* screen. This function can be useful for systems that have written programs to update the *SMTP.ADR* and *SMTP.POD* automatically.



Automatically update alias database

Enable / disable the routine checking on the *SMTP.ADR* file.

NOTE: Internet Exchange *Workgroup Edition* supports a maximum of 100 registered users of the gateway. This number is determined by adding the number of unique cc:Mail users found in the Alias and Rules Based Addressing databases. The gateway will check for the limits when either the User Alias or Rules Based Addressing databases are modified. If the number of users exceeds the maximum allowed, the respective database update will not be made. Manual regeneration of either database which results in an error will result in a dialog box to the administrator indicating the problem. Errors detected during Dynamic Conversion will result in the error message being mailed to the gateway administrator.

Automatically update domain database

Enable / disable the routine checking on the *SMTP.POD* file.

File locations

Specify the locations of *SMTP.ADR* and *SMTP.POD* files.

Use new SMTP.ADR format

Send permission, receive permission and comment fields have been added to the *SMTP.ADR* file format in **Internet Exchange** version 2.0 or later. When routine checking of *SMTP.ADR* is enabled, *SMTP.ADR* will be updated if the database is modified inside the *Configure User* screen. If this option is disabled, these three new fields will not be updated in the *SMTP.ADR* file. Using the old format can be useful for interoperability with **Internet Exchange** versions prior to 2.0.

MESSAGE DATABASE RECOVERY UTILITY

Internet Exchange version 2.0 incorporates a new database backend engine designed to improve gateway performance. Unlike previous versions which stored message status and envelope information in separate DOS files, version 2.0 now stores this information in the message database (*mesg.btr*). For reliability, the message file, which used to only contain the message contents, now also contains redundant envelope information, in the unlikely event that problems are ever encountered with the message database.

If the message database file (*mesg.btr*) should ever get removed or corrupted, the *Message Database Recovery* utility can be used to read the envelope information from the message files and create a new *mesg.btr* database file.

Unlike the utilities described so far, the *Message Database Recovery* utility can only be started from the Program Manager. When **Internet Exchange** was installed, it should have been installed in the **Internet Exchange** program group. Upon execution, the following message will be presented:

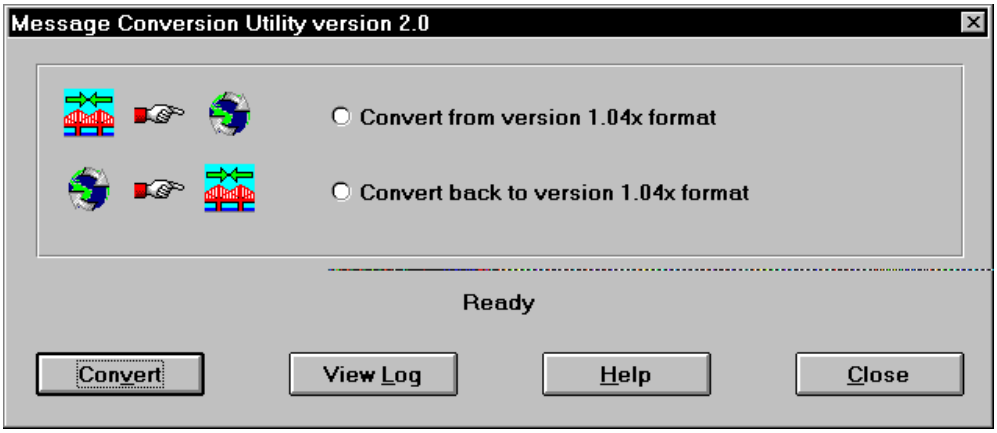
> WARNING!
>
> This utility will remove the original message database file (mesg.btr) and recreate a new one.

To create a new message database, simply depress the *Rebuild* button. Upon successful completion of the operation, the message "*Recovered message database successfully!*" will be displayed in the window status line.

MESSAGE CONVERSION UTILITY

As described above in the section on the *Message Database Recovery* utility, the message file format and location of where status and envelope information has been changed in Version 2.0. The *Message Conversion* utility gives the gateway administrator the ability to switch back and forth between the two formats. This is initially necessary when performing an upgrade from a 1.04x installation. It is also necessary if messages are queued in either the SMTP IN or SMTP OUT queue and the gateway is to be downgraded back to version 1.04x.

The *Message Conversion* utility can be found in the **Internet Exchange** program group and has to be run from the Program Manager. Upon startup, the following screen is displayed:



To perform a conversion, simply choose the appropriate conversion desired, and then depress the *Convert* button. Upon successful completion, a dialog box will be presented indicating the operation has completed.

CHAPTER 11
GATEWAY MANAGEMENT

MESSAGE QUEUES

As detailed in chapter 1, **Internet Exchange** maintains three separate message queues:

cc:Mail Post Office	Stores messages received from the local cc:Mail environment and bound for the Internet.
SMTP IN	Stores messaged received from the Internet by *SMTPD* and bound for local cc:Mail users.
SMTP OUT	Stores messages bound for Internet that have been exported by *CCOUT* from the local cc:Mail Post Office .

Each of the above queues can be viewed separately by selecting the appropriate radio button under the *Select queue* section of **Internet Exchange** System Manager's main screen.

When selected, the entries present in the queue, if any, will be presented in the information area located immediately below the queue selection and logfile dialog boxes. The number of entries in the queue should match the number given in the corresponding queue counter box. Under certain circumstances, such as when the system is under heavy load, the queue entry counters may become unsynchronized with the displayed information. In such situations the System Manager can appear to be unresponsive to user input. This can be considered normal behavior as it is a side-effect of the Windows and VIM implementations.

> ***IMPORTANT***: Under such situations ***DO NOT*** attempt to gain the attention of the system by repeatedly clicking on various boxes. These interrupts are queued for processing when a time slice becomes available and, when processed, may result in an action or actions being taken which were not desired. The responsiveness of the System Manager under these conditions (and the throughput of the system in general) can be improved by turning off *Regular Screen Updates* within the *Options* menu.

LOG FILES

Internet Exchange maintains four levels of logging information. This information can be very useful and may be required in order to determine the causes of certain email and/or network related problems that may occur. By default **Internet Exchange** logs all SMTP session level information. In general, this is both desirable and sufficient. In rare cases more diagnostic information may be required. Logging at this level is too voluminous for general use and significantly reduces system performance.

The information contained in the log file can be viewed by selecting the *View Logfile* button on the *SYSMAN* screen. The default viewer (the *Write* command under Windows 3.1) will be invoked unless the appropriate entry in the *IMA.INI* file has been modified (see the *iecmref.pdf* manual in the installation directory). By default *Write* will first ask if you would like to convert the file (i.e. the logfile) to *Write* format. This is not required, so click on *Cancel* to continue. To exit and return to the System Manager, simply pull down the *File* menu and click on *Exit*.

By default, **Internet Exchange** will log all transactions in the file *ieccmail.log* located in the gateway queue directory. The software will continue to log information until the file size reaches the maximum configured value. At this time, the log file will be renamed and a new *ieccmail.log* file created to continue logging. The naming convention for old logfiles is:

ddmmmnnn.log

Where *dd* is the day of the month, *mmm* is the month, and *nnn* is a three digit number starting at zero, and increasing for each old logfile generated during that day. For example, on March 5th, the first few old logfiles to be renamed would appear as:

05Mar000.log, 05Mar001.log, 05Mar002.log, ...

If the *SendOldLogFile* option is set on the *Configure Advanced Options* screen, old logfiles will be emailed to the local postmaster. A *KeepOldLogFile* option on the same screen controls whether old logfiles will be kept on disk or deleted.

> ***IMPORTANT***: if not deleted, the logfiles will continue to accumulate until manually archived or removed by the gateway administrator. If left unattended, they can grow to consume a significant amount of disk space.

WINDOWS RESOURCE TRACKING

There are three bar gauges on the System Manager screen indicating the state of various critical Windows resources. This allows the administrator to keep an eye on whether **Internet Exchange** is likely to run out of resources in the near future. The gauges display green normally, but change color when space is running low. When less than 50% of a resource is free, the display will change to yellow. When less than 10% is free, the display will change to red. Additional resources should be freed up as soon as possible to ensure continued operation of the gateway. When disk space

is low, files need to be deleted or moved to other disks. When either of the other two resources are low, close all other applications to free up additional resources.

To change how often resource state checks are made, change the *system schedule* option in the *Configure Schedules* screen.

Free GDI

This gauge displays the free Windows GDI resources. This is a small set of internal Windows resources related to graphics that are critical to the operation of **Internet Exchange**.

Free Resource

This gauge displays the free Windows USER resources. This is another small set of internal Windows resources related to the user interface that are critical to the operation of **Internet Exchange**.

Free Disk Space

This gauge displays the free disk space for the queue directory drive.

AUDIBLE WARNINGS

Whenever the SYSMAN bar gauges enter the critical red state, **Internet Exchange** will emit an audible tone to alert the administrator, who might be busy doing other things. This tone will be generated when less than 10% of GDI / User resources are available or when there is less than 5MB of disk space left in the queue directory. The administrator may then check the gateway display to find out which resource needs attention.

A short beep from the PC speaker can be heard when the logfile has reached it's maximum size and is being renamed. If another application, such as an editor or text viewer has the current logfile open and locked, the gateway will continue to beep until the condition is cleared. This condition will not cause any harm to the system, however some log messages may be missing from the logs during the time the problem persists.

The audible warnings can be manually disabled by setting *[Options] DisableAudibleWarning=YES* in the IMA.INI file.

MESSAGE FUNCTIONS

Circumstances can arise which require administrator intervention. It may be necessary to delete, forward, or inquire about the status of a message. The message function buttons provide an interface giving the administrator complete control over all messages passing through the gateway. Note that these actions can only be performed on messages that are waiting to be delivered. Once delivered to either cc:Mail or the Internet, they are out of reach.

It is generally desirable to make sure sync mode is off and set the queue run time to some large number (e.g. 1000 minutes) before attempting operations on a given message. Otherwise, the queue may be processed too fast to allow individual messages to be selected.

Deliver

Selecting this option forces immediate delivery of the selected message.

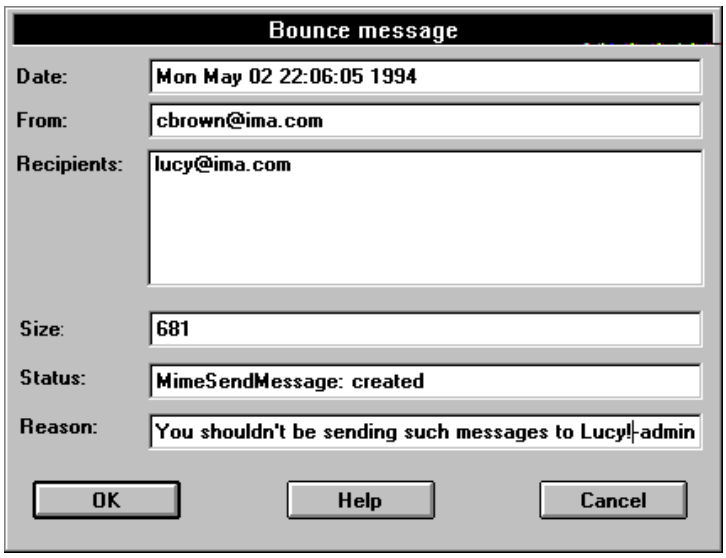
Forward

This option presents a screen allowing the administrator to forward a selected message to one or more recipients. For example:

Once the forward operation is selected, the gateway administrator needs to enter the Internet address in the *Forward To:* field and then select the *OK* button to perform the forwarding operation.

Bounce

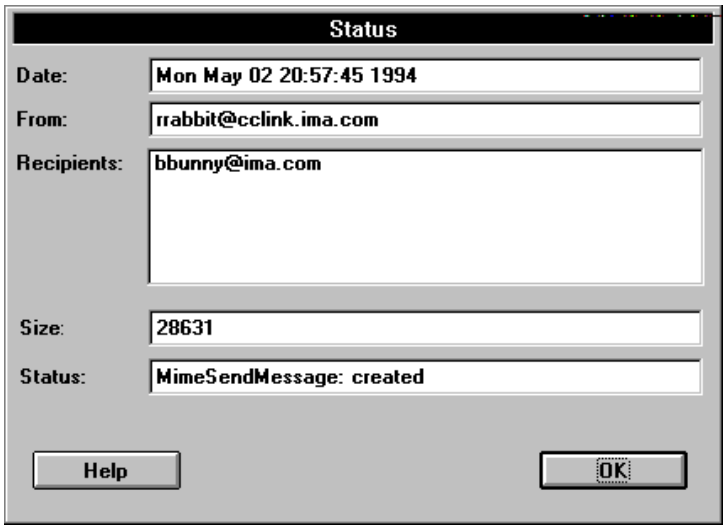
The bounce option lets the administrator return a message to the sender with an optional explanation.



Once the bounce operation is selected, the administrator can enter a reason for returning the message in the *Reason:* field. The message will then be bounced after the *OK* button is selected.

Status

The status of a message can be determined either by selecting a message and then clicking on the *Status* button or by double-clicking on the desired message.



Delete

The *Delete* option deletes a message from the message queue, after first asking for confirmation. This protects against accidental deletion of a message.

After the delete action is verified, the system responds with confirmation of the deletion.

APPENDICES

APPENDIX A
TECHNICAL SUPPORT
FREQUENTLY ASKED QUESTIONS

This document lists the most common technical support problems, and how to solve them. An updated version can be found in the following locations:

http://www.ima.com/support/ccmail/techfaq.html
ftp://ftp.ima.com/pub/support/ccmail/techfaqwri

Most problems are related to incorrect installation, so make sure that the gateway and related software has been installed correctly.

Note to Novell users

Novell LAN Workgroup is NOT supported at this time. For further information, contact your Novell supplier.

Users of Novell's LAN Workplace 4 for DOS need to download patches before installing **Internet Exchange**. These are available by anonymous ftp at:

ftp://ftp.novell.com/pub/updates/unixconn/

The directories lwdos41 and lwdos42 contain updates for those versions, as well as an index file describing the available patches. Another useful site on the Web is:

http://netwire.novell.com/FileUpdt/

An alternative location is the Novell BBS at 408-649-3443 in the United States. Which patches are required depends upon the version of LAN Workplace that you are running.

After the Winsock patch has been applied, make sure that the ip_address entry in NET.CFG matches an entry in the host file for the local host. It should be the FQDN, not an alias.

LICENSING

How do I upgrade from an interim license to a permanent license?

Shutdown the gateway and run the License Update program. Select the *permanent* radio box and enter the License Key provided by your supplier. No expiration date is needed. Then hit the Update button. A message should appear indicating a successful update has been performed. If not, contact your supplier for further information.

Admin displays the message: Invalid License Key - check the support FAQ for further information.

If the License Update program did not complete successfully, rerun it using the information provided by your supplier. Also, if you are planning to change the gateway hostname and domain, you must contact your supplier for new licensing information before you make the change. The license key is based upon this information. The supplier will provide new license information. Once you receive this, change the gateway hostname and/or domain by way of Setup, and rerun License Update to enter the new license information.

If the error message persists, contact your supplier for further information.

Admin displays the message: Invalid Expiration Date - check the support FAQ for further information.

See the above explanation.

Admin displays the message: Invalid License - check the support FAQ for further information.

See the above explanation.

Admin displays the message: License has Expired - check the support FAQ for further information.

See the above explanation.

Message throughput is low when there are large numbers of messages in the queues.

This happens because the gateway spends a great deal of time regularly redrawing the message queues. The *regular screen updates* option should be turned off.

WINSOCK

When I startup the gateway, I get a message saying that WINSOCK.DLL cannot be found.

This is because Winsock has not been installed correctly on the system. Some stacks have a separate procedure to follow to install Winsock. Call your stack supplier for further information.

Another thing to check on is that WINSOCK.DLL is in the path. If not, the above message will appear. Add the TCP/IP stack directory to the path, preferably by editing the autoexec.bat file. This directory is usually where the WINSOCK.DLL file is located.

Another way to test the local Winsock stack is to try a few public domain Winsock programs. These are available by anonymous ftp from various sites on the Internet. One example is ftp.cica.indiana.edu, in the directory pub/pc/win3/winsock. This directory contains great many Winsock applications. Check the FAQ for alt.winsock for other locations. Please contact IMA support if you are unable to find either the FAQ, or the programs. If these Winsock programs do not run correctly, then the Winsock installation is incorrect. If they run fine, then the problem is with the gateway.

SMTPD produces a General Protection Fault.

Follow the above instructions to make sure that Winsock is installed correctly. Also make sure that at least 500kb of low memory is available, as well as at least 4MB of RAM.

The SMTP daemon and/or client exit, logging a message *WSAAsyncSelect failed.*

This can occur if Winsock has not been installed correctly. Read the answer to the first Winsock question to make sure that installation is complete.

Another cause of this error is lack of memory. Internet Exchange requires at least 500kb of low memory. If this is not available, the above message is sometimes displayed. The solution here is to load as much as possible into high memory at system startup. This should allow all programs to load correctly. If any other programs are running with the gateway, close them to free up more memory.

The gateway displays the message *NOVASYNC.EXE cannot be loaded.*

This occurs with an incorrect installation running Novell LAN Workplace for DOS. Check that Winsock has been installed correctly. It should be noted that Winsock cannot be installed just by copying over WINSOCK.DLL. Follow the instructions given in the first Winsock question to install Winsock for Novell LAN Workplace.

The SMTP daemon and/or client fails, logging *gethostname failed.*

First, check the above to make sure that the TCP/IP stack has been configured with the correct local hostname. Also make sure that the local stack has the correct DNS servers configured.

Make sure that the local hostfile is configured into the gateway correctly, under Configure Connection. Make sure the local hostname is listed in the hostfile, with the fully qualified domain name (FQDN) listed first.

Under Wollongong Pathway, a Winsock error of WSAENETDOWN(10050) is logged.

This usually happens if the TCP stack has not been correctly installed. It means that the network is down. i.e. WSOCKCB.EXE cannot be loaded. Reinstall the stack to fix this problem.

Using FTP Software v2.2, v2.3 or v3.0, under heavy loads, the gateway either hangs, crashes Windows, or reboots.

This is a known problem with FTP Software stacks. We do not recommend use of any of these stacks.

SMTPD or SMTPC fails, logging WSAEMFILE(10024)

This error message indicates too many open files. Close all other running applications to free up some file resources. Another option is to adjust the FILES parameter in the config.sys to allow more open files.

SMTPD or SMTPC fails, logging WSAEADDRINUSE(10048)

This occurs when another SMTP daemon is running. Close the other process and restart SMTPD. It might also occur if SMTPD has just crashed. In this case, the TCP stack needs to be unloaded and reloaded. This procedure varies depending upon which stack you are running. Check with the TCP/IP documentation for further information. If this does not work, then reboot the machine.

SMTPD or SMTPC fails, logging WSAENETDOWN(10050)

This error indicates that the network is down. Usually this is due to an incorrect Winsock installation. It also may happen if a network connector has been removed.

SMTPD or SMTPC fails, logging WSAECONNABORTED(10053)

The connection was aborted remotely. This is usually due to an error condition at the other end, and requires no further action. The message will be retried later by *SMTPC.*

SMTPD or SMTPC fails, logging WSAECONNRESET(10054)

The connection was reset remotely. Nothing needs to be done in this case.

SMTPD or SMTPC fails, logging WSAENOBUFS(10055)

This error message indicates that no buffer space is available. Reconfigure the TCP/IP stack to increase the number and size of buffers. The machine will need to be rebooted for the changes to take effect. Under Novell, the following values should be used in NET.CFG:

```
link support
  buffers 15 1500
  mempool 32000
```

If running on a token ring, increase the number of tcp sockets and udp sockets configured.

SMTPC fails, logging WSAECONNREFUSED(10061)

The remote site refused to talk to SMTPC. This usually happens when the remote site has some sort of problem. Many nameserver machines refuse SMTP connections, which might be the problem. Try to contact the remote machine by telnetting to port 25, the SMTP port. This will probably be refused also. Contact the administrator of the remote site to find out why this is happening.

SMTPC fails, logging WSAEHOSTDOWN(10064)

The remote host is down. The message will be retried later by SMTPC.

VIM

All imported messages from the Internet have a zeroed year in the date.

This is due to an old version of the VIM libraries. Internet Exchange requires at least VIM Version 2.07 to solve this problem. Download the latest version from the following locations:

The cc:Mail BBS can be reached at 415 691 0401 within the USA. Call with any asynchronous package, with parameters: 8-N-1.

To obtain the files by anonymous FTP, the current URL is:

ftp://ftp.ccmail.com/pub/comm/ccmail/dev_tools/vdlwin.zip.

Install the new VIM libraries in the same directory as cc:Mail, and make sure that this directory is in the path. Otherwise, Windows will be unable to load these libraries at runtime.

If dates are still not working correctly, make sure there is only a single copy of the VIM libraries on the system. Windows will load the first one it finds, and it might find the old version first. The easiest way to do this is to search for VIM.DLL by doing a *DIR /S VIM.DLL* command from the root directory under MS-DOS, using *File Search* from the File Manager under Windows 3.1 or using Windows Explorer under Windows 95.

The programs CCIN and/or CCOUT fail with *VIMEnumerateMessage failed* message.

This happens when cc:Mail gets out of sync internally. Shutdown the Post Office and run *chkstat* followed by *reclaim*.

Chkstat and *reclaim* should be run regularly to increase performance of both cc:Mail and the gateway.

The following VIM error is logged: VIMSTS_INSUFFICIENT_MEMORY

The system has run out of memory. Make sure that there is at least 4MB of RAM available, as well as at least 500kb of low memory. Shutdown any unnecessary programs that are running. Free up some low memory by loading drivers high or using tools such as memmaker, QEMM, etc.

The following VIM error is logged: VIMSTS_INVALID_CONFIGURATION

The local cc:Mail system has not been setup correctly. Reinstall cc:Mail and make sure the latest VIM libraries are loaded.

The following VIM error is logged: VIMSTS_INVALID_PASSWORD

The cc:Mail Post Office password is incorrect. Please enter the correct one by way of Configure Post Office.

The following VIM error is logged: VIMSTS_NAME_NOT_FOUND

A message was addressed to a cc:Mail recipient who does not exist. This may happen when an old user has been deleted. Be sure to delete any entries for that user from the alias file, by way of Configure Users.

The following VIM error is logged: VIMSTS_WRITE_FAILURE

A file could not be written on the system. This will usually happen if the gateway machine runs out of disk space. Delete some files to free up some disk space, and retry the operation.

The following VIM error is logged: VIMSTS_CONTAINER_CORRUPT

The local cc:Mail Post Office is corrupt. Run *chkstat* and then *reclaim* to restore the Post Office to a useable state. These programs should be run regularly to ensure that the cc:Mail database remains consistent. The required interval might be weekly, or as often as every day, depending upon message traffic.

Sometimes CCIN logs the following message: *VIMSetMessageRecipient aux address failed.*

This happens when importing a message with a large number of Internet recipients. There is a 4kb internal limit for cc:Mail headers, and this message will be logged if that limit is exceeded. To ensure all the addressees are saved, set the option to include rfc822 headers.

MISCELLANEOUS

How can I get rid of extraBcc's that plague our inbound mail?

Bcc's are generated by **Internet Exchange** as an attempt to convey information present on the Internet side of the gateway, that has no equivalent in cc:Mail - the envelope recipient. A full explanation requires a few preliminary words.

The most commonly accepted models for electronic mail (RFC821/822 in the Internet, X.400 in the OSI world, UUCP, etc), are based on a clear-cut distinction between envelope (used by the Mail Transfer Agents, or MTA's) and headers (intended for humans handling mail through User Agents, or UA's). In theory, an MTA should never touch the headers. The envelope consists of a "sender field" and one or more "recipient fields", or the recipient list. In Internet mail, usually the envelope recipient is derived by the originating UA from the "*To:*", "*Cc:*", and "*Bcc:*" headers and passed onto the first MTA.

As a message passes from the original MTA where the message was submitted to **Internet Exchange**, it may have passed through many other MTA's along the way. At each of these "stops" along the path, the message may have encountered either autoforwarding (user or system level forwarding), and/or distribution list expansion. The effect of either autoforwarding or DL expansion is the replacement of one or more elements of the envelope recipient list with one or more new addresses. By the time the message finally arrives at the final MTA (**Internet Exchange** in our case), the envelope recipient list may differ considerably from the original envelope and the message header recipients.

When **Internet Exchange** receives a message from the Internet, it has to face one of the hard facts in a gateway's life: different mail models on the two sides. In our case, cc:Mail does not distinguish between envelope and header recipients - the only way to deliver a cc:Mail message is to place the recipient's name into a "*To:*", "*Cc:*", or "*Bcc:*" cc:Mail recipient field. **Internet Exchange** will attempt to deliver to all cc:Mail recipients present in the envelope recipient list. In order to accomplish this, it needs to place the recipient addresses into the appropriate cc:Mail recipient fields. It does this by comparing the envelope recipient list to the original message header recipients. All recipients that can be matched in this manner are placed in the appropriate cc:Mail "*To:*", or "*Cc:*" field. All others are categorized as "*Bcc:*" recipients.

A common configuration for **Internet Exchange** installations is for the gateway to be hidden behind one or more mail relays. When mail is sent out from the gateway, it carries the name of the mail relay host. When this mail is replied to, the messages first go to the mail relay, where they are forwarded to **Internet Exchange** (with the recipient list modified). If the name of the gateway machine is *iegate.jade.net* for instance, the mail relay might be *jade.net*. A mail message sent to "user@jade.net", would have a header recipient field and original envelope address of "user@jade.net". After the message is processed by the mail relay (*jade.net*), the envelope address is rewritten to be "user@iegate.jade.net". When **Internet Exchange** processes this message, the envelope and header addresses do not match, resulting in a cc:Mail "*To:*" header of "user@jade.net at INTERNET", and a "*Bcc:*" which contains the envelope address. For situations like this, the proper solution is to declare the domain "jade.net" as local by adding it to the Alternate host/domain name list.

In the above example, why isn't the mail relay name automatically added to the Alternate host/domain names list?

The above solution will work well in the case where only cc:Mail users are being represented by the mail relay. Let's suppose that some users of jade.net have accounts on both the cc:Mail and the Internet sides (e.g., on some UNIX host). Now let's suppose that a generic local cc:Mail user wants to send a message to the mythical Joe User, both at his UNIX and cc:Mail accounts. The most intuitive thing to do is addressing the mail to "user@jade.net at INTERNET", so that the gateway can convert it into Internet mail for *user@jade.net*, deliver it to the mail relay, which will either deliver it to a local UNIX mailbox or continue to forward it to another machine for final delivery. Unfortunately, if *jade.net* is listed under the Alternate host/domain names list, this will NOT work. The reason is because the gateway's SMTP delivery module (SMTPC), before attempting a delivery to the Internet, always checks whether the domain is a local one.

In this case instead of connecting to the SMTP listener on the same gateway machine, it sends the message back to the inbound converted (CCIN) through the local system.

Remote systems or mail relays that are required to accept mail for recipients different from the **Internet Exchange** gateway should never be listed in the Alternate host / domain list.

Messages bounce around between the SMTP in and out queues, never going anywhere.

This can happen if the addressing options have been setup incorrectly. e.g. sometimes mail addressed to:

user@domain.com

comes into the gateway system, but the *use host name in addresses* option is turned on. This means that only messages addressed to

user@host.domain.com

will be accepted. Thus, CCIN will reroute the message back out of the gateway, and it will probably be continually routed back in and out. There are several ways of solving this problem.

The first is to turn off the *use host name in addresses* option. This will ensure that the above message can be delivered locally instead of rerouted out of the gateway again. Then add domain.com to the alternate host name / domain list under Configure Connection.

Another solution would be to keep this option on, and set the gateway hostname to be 'domain' and the domain to be 'com'. Although counterintuitive, it will result in the message being delivered.

Another option would be to make sure that all mail is addresses to the second format above.

Another cause for this problem to occur is when the mail relay host is the same as the local hostname. This is not very useful, and should be changed to point to a smart machine to which the gateway forwards mail as a last resort.

SMTPC cannot resolve the mail relay hostname.

This might happen if the mail relay hostname is misspelled, or does not exist. After checking this, make sure there is an entry in the hostfile for the mail relay host, including a FQDN.

Incoming uuencoded files aren't being decoded.

This sometimes happens when a message is imported. Often, the cause is that it's part of a MIME message, and the Content-Transfer-Encoding is set to something other than x-uuencode. In this case, it will NOT be decoded. The Content-Transfer-Encoding header must be set to x-uuencode for automatic decoding.

Messages are going out ok, but none are coming into the gateway.

First, goto Configure Gateway to see if the gateway isn't in either send only or configure only mode. Make sure that the outside world knows to send messages on to the gateway. Either add an entry for the gateway machine to external host files, or update remote DNS servers so that they know about the gateway machine. Try to ping the local machine to check whether address lookup works.

If messages are stuck in the SMTP IN queue, check to make sure CCIN is being run regularly. Check to see that the CCIN interval isn't too large. Turn on sync mode for CCIN for best effect. Check in Configure Post Office that the Internet Post Office name is set correctly. There needs to be an entry in the cc:Mail Post Office for the gateway, which matches the Internet Post Office name.

Messages are coming in ok, but none are going out of the gateway.

First, goto Configure Gateway to see if the gateway isn't in receive only or configure only mode. Check to see that the CCOUT interval isn't too large. Set sync mode for CCOUT for fastest message export. Try running the cc:Mail queue manually. Check the Internet Post Office name, as described above.

If messages are becoming stuck in the SMTP OUT queue, check to make sure that host addresses are being resolved correctly. If Name Resolution in Configure Connection is set to host access only, make sure that all desired destination hosts are in the hostfile, and that the gateway points to the right place in Configure Connection. If using DNS, make sure that the DNS server addresses in Configure Routing are correct. Try using *wshost* and *ping* to see if the DNS servers are working. Also, make sure that there is an entry for the mail relay host in the hostfile, in case all other methods of address resolution fail.

SMTPC logs that it can't find a hostname, even though it is listed in the host table

Make sure that the gateway is looking at the correct host table. i.e. check in Configure Connections. If this is ok, check the host table itself. Sometimes the last entry doesn't have a CRLF at the end of the line. In this case, SMTPC will not find that entry. If this is the case, just add a final CRLF at the end of the host table.

APPENDIX B
WINDOWS 95 DIALUP NETWORKING SCRIPTS

The following information is copyright © Steve Jenkins 1995 and used with permission. The most uptodate version is available at:

http://www.windows95.com/connect/dscript.html

In order to automate the repetitive manual steps that must be completed to log into many ISPs around the world, more and more users are looking to scripts as a way to facilitate their login process. This section of discusses how to create scripts and how to use the Windows 95 built-in scripting tool to attach a script to a Windows 95 dial-up session. In addition, I have also created a page with example scripts that work with real-world connections.

Note about the Microsoft Scripting Tool

Many scripting users have been having trouble getting some of the Microsoft scripting commands to work. This is because there are two versions of the scripting tool available. The basic scripting tool (which ships on the Windows 95 CD-ROM) supports simple scripts only, like the sample script provided in this tutorial. The default scripting tool should be sufficient for the majority of script users.

Using the more advanced scripting commands requires the scripting tool available in Microsoft PLUS!, which is available at your local software retailer. Any script that uses the integer command requires the Microsoft PLUS! version of the scripting tool.

Step 1: Create a Dial-up Connection

Before writing your script, make sure that you have a working dial-up connection to your Internet provider. If you do not, use the information on the Windows95.com TCP/IP Setup Page to help you set one up. Once you have a working connection, proceed to step 2.

Step 2: Create a Login Script

Using a text editor (Notepad will work fine), create a script that will issue the commands necessary to log you into your Internet provider. If you have a PPP connection, many providers do not require manual log in steps - your username and password in the Connect To dialog box will be sufficient, and you won't need dial-up scripting at all. However, all SLIP/CSLIP connections, some PPP connections, and any other connection that requires menu selections, advanced

input, or that does not strip information from the initial dialog box will require a script.

If you have a PPP connection that you think requires a script, try putting *ppp:your_username* in the username text area in the Connect To dialog box. Some providers will accept this to initiate a PPP connection. If that doesn't work, try a script. In most cases, you will most probably be able to modify an existing script to connect successfully with your provider. Sample scripts are available here.

All scripts must begin with the following line:

proc main

and end with:

endproc

These are the commands to tell Windows 95 to start and stop the script. Use the *waitfor* and *transmit* commands to wait for certain information from your provider (such as a login: prompt or a password: prompt) and to transmit your username, password, and any other necessary information. The variables *$USERID* and *$PASSWORD* will send the username and password entered in the Connect To dialog box at the beginning of the dial-up session. To send a carriage return to you provider, use a *^M*. To wait for any amount of time, use *delay* followed by the number of seconds. You can put comment lines in your scripts by beginning the line with a semicolon (;). For example: a simple script that starts, waits for a login: prompt, sends your username, hits ENTER, waits for a password prompt, sends your password, hits ENTER then ends, would look like this:

```
;This will begin the script

    proc main

;Enable the following to delay for 3 seconds first to
;allow host time to send initial characters (not needed by many ISPs).

    delay 3

;Sometimes, ISP's need a carriage return to initiate the login process.
;If your ISP requires this, uncomment the following line:

    ;transmit "^M"

;Wait for the login prompt before entering the user ID and carriage return
;(I left off the first letter since login is case-sensitive)
;The $USERID variable is taken from the dial-up connection dialog box

    waitfor "ogin:"
    transmit $USERID
    transmit "^M"

;Enter your password (I left off the first letter since login is case-sensitive)
;and send a carriage return

    waitfor "assword:"
```

transmit $PASSWORD
transmit "^M"

;Finish the script!

endproc

In fact, the above is the exact script I use to connect to my provider! If you use the integer command in your script, you will need the version of the Dial-up Scripting Tool available in Microsoft PLUS!.

If your provider requires PPP callback, try inserting these lines into your script:

delay 1
transmit "++++"
delay 1
transmit "at&c0q0o^M"

Once you're finished with your script, save it in the *\Program\Files\Accessories* folder, with a file extension of scp (i.e. *ppp.scp*).

Step 3: Verify that the Dial-up Scripting Tool is Installed

Press the Start button under Windows 95, select Programs, then Accessories. If the Dial-Up Scripting Tool shows up, you're ready to proceed to step 4.

If it is not there, and you have the Windows 95 CD, press the button, select Settings..., then Control Panel. Double-click the Add/Remove Programs icon. Select the Windows Setup tab, then click on the Have Disk.. option. Assuming your CD-ROM drive is E:, enter the path of e:\Admin\Apptools\Dscript. Press OK and the SLIP/CSLIP drivers and Dial-up Scripting Tool will be installed.

If you have Windows 95 on floppies, you can download the Dial-up Scripting Tool and SLIP drivers directly from Microsoft (be aware, however, that their server is VERY busy). Then install the Dial-up Scripting Tool as shown in the above paragraph.

Step 4: Attach Your Script to a Dial-up Profile

Press the Start button under Windows 95, select Programs, Accessories, then Dial-Up Scripting Tool. You'll see the utility's dialog box.

Your current dial-up profile(s) will be listed in the text area on the left. Select the profile for which you wrote the script, and then press the Browse button. If you saved your script in the Accessories folder, it should appear in the dialog box. Select it and press Open.

If you'd like to troubleshoot your script (recommended for the first time through), select the Step through script option. Upon connection, this will allow you to "step through" each line of your script and see the result in a terminal screen. When your script is working properly, select the Start terminal screen minimized option to keep the script window minimized when you connect.

Press the Apply button (it's best to keep the Dial-up Scripting Tool dialog box open until you finish troubleshooting your script) and get ready to try your script! When you are certain it works, you can press OK to close the Dial-up

Scripting Tool. It does not have to be open when you connect for the script to work.

Step 5: Connect and Troubleshoot your Script

IMPORTANT! Before you connect and use your script, go to My Computer, double-click Dial-up Networking, select your dial-up profile, click the right mouse button, and select Properties. Underneath your modem (in the Connect using section), press the Configure button. Select the Options tab and make sure that in the Connection Control area, NEITHER OPTION IS SELECTED. Even though you probably had the Bring up terminal window after dialing option selected previously, the Dial-up Scripting Tool opens a terminal window anyway. Leaving this option checked will cause your script to fail.

Using Dial-up Networking, connect to your Internet service provider. Make sure your username and password are entered into the dialog box, since your script will need these variables to connect. If you chose to step through the script, a terminal window will appear upon connect and let you step through your script by pressing F7. Watch the result closely to track down errors in your script. You can press F3 during this process to cancel at any time.

Once your script connects reliably, turn off the Step through script option in the Dial-up Scripting Tool dialog box. You can also close the Dial-up Scripting Tool. It does not need to be open for the script to run. It will run automatically with your dial-up connection as long as it is attached properly.

Example Scripts

I have put together a collection of sample scripts that work with Internet Service Providers around the world. You can find them in my Sample Scripts page. All of them are easily modifiable to work with your particular provider.

APPENDIX C
MIME

INTRODUCTION

MIME, which stands for Multipurpose Internet Mail Extensions, is a standard that extends the functionality of basic Internet mail (RFC-822 message format) to allow additional types of message contents to be transported by Internet mail services.

The previous standards, RFC-821 (SMTP) and RFC-822 (Text message format) were published in 1982 and have been very widely implemented, even among non-Internet mail systems. However, these standards impose a very strict limitation on what can be included in an Internet mail message: the content is limited to 7-bit ASCII text (with lines shorter than 1000 characters). MIME permits this to be extended to include message contents such as:

- Images and graphics files
- Text in non-ASCII character sets
- Text in fonts
- Text with arbitrary-length lines
- Sound and video messages or files
- Multi-part messages
- Binary and application-specific files
- References to documents stored elsewhere

In general, these content objects are (when required) encoded to permit them to travel through existing Internet mail systems. MIME messages are downward-compatible with RFC-822 and will interoperate seamlessly even with mail systems that have not been upgraded to special MIME capabilities. This means that MIME messages can be transported through the 7-bit ASCII path provided by SMTP; that MIME messages will not cause RFC-822 mail user agents to break; and that MIME mail user agents will be able to handle normal RFC-822 messages as well. (It is also possible for sets of hosts that are MIME-capable to communicate using an extension to SMTP that allows some MIME messages to be transported without encoding. This is called ESMTP and is discussed below.)

Conceptually, MIME works by defining some additional RFC-822-type headers that describe the structure of the message. The single message body of RFC-822 messages is extended to multiple message bodies in MIME, each of which can contain a different content type. And each content type can be either transported in native form (if compatible with RFC-821 and RFC-822) or encoded to be compatible.

MIME HEADER FIELDS

The additional header fields defined by MIME are the following:

MIME-Version	Version of the MIME standard used by the message. Currently 1.0.
Content-Transfer-Encoding	Method used to encode the content data for transport
Content-Type	Data type of the contents. There are numerous subtype and option fields
Content-Description	Description of the message body data
Content-ID	Unique identifier (similar to the Message-ID) for message body parts

MIME was designed to be extensible; the set of content types, subtypes, and options is easily extensible, as well as the set of transfer encodings. The MIME standard (RFC-1521) requires that new values for these basic types be registered with the Internet Assigned Numbers Authority (IANA) to prevent confusion and name collision.

When a MIME message contains a single message body part, the main (RFC-822) message header will contain the required Content-Type and Content-Transfer-Encoding headers with the appropriate value. The RFC-822 message body will contain the MIME message body, either as a simple text message, or in the specified encoding. A simple MIME message with a message body with a Content-Transfer-Encoding of "7-bit" (the default) and of Type "text/plain" will simply appear as an RFC-822 message.

When a MIME message has multiple message bodies, however, the Content-Type header in the RFC-822 message header will specify

```
Content-Type: multipart/mixed;        boundary="unique-
boundary"
```

This means that the individual message bodies will be set apart with the string

```
--unique boundary
```

alone on a line. Each message body must then contain its own Content-Type header that defines the type and subtype of the message body part. It is

possible to nest multipart messages (as long as the nested levels use uniquely defined boundaries).

This simple example multipart message consists of two message body parts:

```
From: milo@bloom-county.outland.com (Milo)
To: charlie-brown@peanuts.comic.com (Charlie          Brown)
Subject: Simple example
MIME-Version: 1.0
Content-Type: multipart/mixed; boundary="unique-      1a7GHq5cm"

This is a preamble to the message, ignored by MIME mail user
agents.

By convention a message to non-MIME user agents is placed here
explaining that this is a MIME message.

--unique-1a7GHq5cm

This is part 1 of the message.  There was no Content-type or
Content-transfer-encoding header, so it defaults to type
"text/plain" and encoding "7bit".

--unique-1a7GHq5cm
Content-type: text/plain; charset=US-ASCII

This is part 2 of the message.  It is explicitly typed as
"text/plain".
Its default encoding is also "7bit".

--unique-1a7GHq5cm
This area is the message epilogue, also ignored by MIME mail user
agents since it is beyond the boundary of the last message part.
```

There are multipart subtypes other than "mixed", indicating other ways that the multiple message bodies are to be interpreted.

Content Types

There are 7 top-level content types defined in the MIME standard. Additional supported content types are expected to be included by defining additional subtypes of the top-level types. The seven top-level types and their currently defined subtypes are:

Type	Subtype	Remarks
application	(various)	Binary data specific to an application. A full list of subtypes can be found in RFC-1521.
audio	basic	8-bit ISDN u-law, single channel, 8000 Hz
image	gif	GIF (Graphics Interchange Format)
	jpeg	JPEG (Joint Photography Experts Group) format
message	rfc822	Encapsulated RFC-822 message, with headers

	partial	Partial message, with identifier and part number. Used to fragment and reassemble large MIME messages
	external-body	Reference to a document or file stored elsewhere, with access method and identifier
multipart	mixed	Multiple independent message body parts, separated by unique boundary, designed to be viewed serially
	alternative	Same as multipart/mixed, but each part is an alternative version of the same information; the mail user agent should choose the most appropriate type
	parallel	Same as "multipart/mixed", but the parts are meant to be presented simultaneously, as in a multimedia message (e.g., graphics plus audio)
	digest	Each part is an RFC-822 mail message, collected into a digest
text	plain	Plain, unformatted text; default character set is US-ASCII
	richtext	A simple marked-up formatting language for text, defined in the standard
video	mpeg	MPEG (Motion Picture Expertise Group) format video
x-*typename*	(as defined)	Privately defined content type used between cooperating mail systems; analogous to RFC-822 "X-" headers

The Name parameter

This parameter was optional in the first MIME RFC (RFC-1341) and is deprecated by RFC1521, but is still used by many mail programs. Its purpose is to suggest a filename for the attachment, in case the user decide to save it to disk. When present, it is used by **Internet Exchange** as name of the cc:Mail file attachment created from a MIME bodypart. If absent, a unique filename *MIMEnn.ext* will be generated, *nn* being a small decimal number, and *ext* an extension determined by the MIME mappings or, if these do not help, the default extension *raw*.

CONTENT TRANSFER ENCODINGS

As shown above, a number of the content types possible in MIME are represented in their native format by binary (8-bit) data, which cannot be transported by SMTP or represented in a standard RFC-822 message.

In order to permit binary data to be transported over standard mail systems, MIME supports a number of encoding formats that reduce the binary data to a manner that is acceptable for transport. This is done through the Content-transfer-encoding header field. The values are as follows:

base64	Basic binary encoding — three 8-bit octets are represented as four printable ASCII characters. Lines are limited to 76 characters
quoted-printable	Lines of mostly printable text, with non-printable characters escaped using "=" as a quote
8bit	Unencoded 8-bit data that is line-oriented (less than 1000 characters per line).
7bit	Unencoded, plain 7-bit characters, less than 1000 characters per line. This is the default format.
binary	Arbitrary-format unencoded binary, no specified line length
x-*encoding*	Privately defined encoding used between cooperating mail systems

Note that 8bit and binary are in fact unencoded forms, and generally cannot be used as such using SMTP transport, except as noted below. 7bit is also unencoded, and is the default format for simple text messages.

As noted above, the set of subtypes is expected to grow in the future; consult the STD version of the MIME RFC for the currently defined subtypes.

OTHER MIME HEADERS

MIME also defines two other headers: Content-ID and Content-Description. These fields are optional. Content-ID is used to label a message body, where it may be useful to refer to it elsewhere, such as in another header field or

message body. Content-Description merely associates a text description of the subject or contents of a (possibly non-textual) message body; this could be used by a mail user agent to display a caption for an image, for example.

EXTENSIONS TO SMTP (ESMTP)

While MIME provides a method of encoding arbitrary types of message contents for transport through Internet standard mail systems, it is also desirable for systems with the capability of exchanging binary data in mail messages to be able to do so (to avoid the overhead of encoding and decoding of 7-bit representations of binary data), in a way that is compatible with the MIME standard.

The extensions to SMTP defined in RFC-1651 are optional and are not required to be implemented by Internet mail systems. Basically, they provide a framework for systems that use various enhancements to be able to identify themselves to each other and discover which extended capabilities they mutually possess. This is similar to the options negotiation in the TELNET protocol. The extended capabilities discussed in the RFC include a maximum message size negotiation and a negotiation for use of 8-bit MIME transport. The 8-bit MIME transport itself is specified in RFC-1426.

If an ESMTP-capable SMTP client wishes to initiate an ESMTP session with a server, it uses the new EHLO command (instead of HELO) to identify itself at the outset of the connection. An ESMTP-capable server will respond to the EHLO command with a response consisting of a list of extended commands (beyond the set required in RFC-821) that it supports. Obviously, early non-ESMTP-capable servers will not even recognize the EHLO command, and will return an "unknown command" error. In this case the client will know that the server is not ESMTP-capable, and the session will continue as a normal SMTP session.

The initial set of extended commands defined by RFC-1651 are:

SEND	Send message to terminal
SOML	Send message to terminal, or mail
SAML	Send message to terminal and mail
EXPN	Expand mail list or alias
HELP	Return a help message
TURN	Client and server exchange roles

Note that all of these were originally defined in RFC-821 as SMTP commands, but were not widely implemented. This standard makes them optional extensions.

RFC-1426 defines an optional extension named 8BITMIME, which can be used if the ESMTP server is able to support 8-bit binary mail transport. The client, upon receiving the 8BITMIME keyword in response to a EHLO command, may send an 8-bit MIME message by extending the SMTP MAIL command with the BODY parameter (with a value of 8BITMIME). Without the BODY parameter, or if it has the value 7BIT, the type will default to 7-bit ASCII text.

The following example from RFC-1426 shows use of 8BITMIME in an ESMTP session:

```
Server: <wait for connection on TCP port 25>
Client: <open connection to server>
```

```
Server: 220 dbc.mtview.ca.us SMTP service ready
Client: EHLO ymir.claremont.edu
Server: 250-dbc.mtview.ca.us says hello
Server: 250 8BITMIME
Client: MAIL FROM:<ned@ymir.claremont.edu> BODY=8BITMIME
Server: 250 <ned@ymir.claremont.edu>... Sender and 8BITMIME ok
Client: RCPT TO:<mrose@dbc.mtview.ca.us>
Server: 250 <mrose@dbc.mtview.ca.us>... Recipient ok
Client: DATA
Server: 354 Send 8BITMIME message, ending in CRLF.CRLF.
            ...
Client: .
Server: 250 OK
Client: QUIT
Server: 250 Goodbye
```

APPENDIX D
TCP/IP

WHAT IS TCP/IP?

TCP/IP (which stands for Transmission Control Protocol/Internet Protocol) has become the most widely used network protocol suite in the world. It is the basis for the global Internet, which by early 1994 consisted of approximately 22,000 connected networks with 1.75 million hosts and about 13 million users. While the Internet can support a number of other networking protocols, TCP/IP is used by the vast majority and forms the basis for Internet routing and addressing.

A network protocol suite is a set of individual protocols (communication standards) that have been designed to work together to support a range of network applications. Each protocol in the suite handles a different part of the entire networking scheme. So called low-level protocols handle matters such as routing or creating virtual circuits, while higher-level protocols may handle data presentation or specific network applications like e-mail or file transfer. A protocol suite is sometimes referred to as a "stack" — a reference to diagrams that illustrate the suite as a stacked set boxes in which protocols performing similar functions are grouped into layers.

TCP/IP is now supported on almost every hardware platform and operating system. In the MS-DOS and Microsoft Windows user community, acceptance was initially slow because of the availability of less complex, turnkey solutions. However, there are now a number of easy-to-install commercial packages that implement TCP/IP for DOS and Windows systems, and are compatible with the WINSOCK standard. Some of these include Novell LAN WorkPlace for DOS, Frontier Technology's Super-TCP, and The Wollongong Group's PathWay.

A complete TCP/IP implementation can be thought of as a "stack" with five levels, of which only the top three are provided by the TCP/IP suite itself.

The Physical layer is essentially merely the actual cabling (or wireless carrier) that carries the electrical signals that are recognized by network equipment as data. It is not part of the protocol suite, but TCP/IP networks are carried on a very wide variety of physical carriers, including twisted-pair cable, coaxial cable, fiber-optic cable, microwave, etc.,

The Data Link layer, sometimes (confusingly) called the Network layer, are a collection of low-level protocols appropriate to various physical cabling schemes. Examples include Ethernet protocols for various coaxial cable and twisted-pair schemes, FDDI for fiber-optic cabling, X.25, V.35 and HDLC for high-speed serial lines, ISDN over circuit-switched telephone lines, token-ring and broad-band protocols, etc. These are also not part of the TCP/IP suite itself, and can be used (often simultaneously) by other network protocol suites, such as NetWare IPX/SPX, DECnet, etc.

The Internetwork layer is the most important to the TCP/IP suite. This is where overall Internet addressing and routing are handled by IP (Internet Protocol).

The Transport layer is concerned with data integrity and consistency. TCP/IP provides two protocols -- TCP (Transmission Control Protocol), which assures a reliable communication stream between processes on two network hosts, with retransmission of lost data, presentation of data packets in the correct order, etc.; and UDP (User Datagram Protocol), which simply transmits packets of data without guarantee of delivery or ordering.

The Application layer is where protocols that implement specific network services are grouped. Most complex and interactive applications use TCP, which will handle the details of reliability and ordering of the communications stream. Examples of TCP-based application protocols include Telnet (the virtual terminal protocol), FTP (File Transfer Protocol), and, of course, SMTP (Simple Mail Transport Protocol). Examples of UDP-based applications include Domain Name System (DNS) queries and the Sun Network File System (NFS) which is the most widely used standard for file sharing on workstation networks.

IP, TCP, and UDP, along with the supplementary protocols used for network control and routing in TCP/IP, will be described below.

INTERNET ADDRESSING

Each system (known as a "host") connected to the Internet has a unique IP address. (Actually, hosts may have more than one address, since they may be connected to more than one physical network, and addresses are actually assigned to the network interface rather than the host CPU.) Unlike addressing schemes for local-area networks, or hardware-based address schemes, IP addresses are assigned by a central numbering authority. This is to ensure the uniqueness of IP addresses in the global Internet, and to make sure that packets will be routed to the correct site network from other parts of the Internet. Obtaining IP addresses for your Internet-connected systems is handled in different ways in different organizations. Some large organizations have been assigned a large "chunk" of IP addresses that are allocated internally by an administrative authority. Other organizations use addresses allocated to them by their Internet service provider. Still others will need to arrange for their own block of IP addresses directly. This manual assumes that you have already obtained an IP address assignment for the systems on which you will operate **Internet Exchange** gateways.

IP addresses are 32 bits long, divided into four 8-bit segments known as octets. Addresses are conventionally written in "dotted-decimal" notation, with each octet expressed as a decimal number, with dots separating each octet. (You may see IP addresses written in hexadecimal or undotted decimal in some programs, mostly involved with installing network adapters or drivers. This is relatively uncommon, however.) The leftmost octet of the address are the high-order 8 bits of the address, followed by the next 8 bits, and so on, with the rightmost octet being the lowest-order 8 bits of the address.

Examples of IP addresses are 192.12.17.2, 36.3.150.9, and 190.9.200.254.

IP Address Classes

Since the Internet is not a single network but a "network of networks," part of the IP address designates a *network number* and part a *host address* on that network. In an attempt to strike a balance between networks of different sizes on the Internet, the IP addressing scheme defines several *classes* of IP addresses. The underlying idea of address classes was to efficiently use the IP address space by recognizing that there will be a very small number of huge networks, a reasonably large number of medium-size networks, and a huge number of very small networks. In addition, provision has been made for two special types of addressing, broadcasts and multicasts.

Class	Address Range	Number of networks in class (theoretical)	Possible hosts per network (theoretical)
A	0.0.0.0 — 127.255.255.255	128	16,777,216
B	128.0.0.0 — 191.255.255.255	16,384	65,536
C	192.0.0.0 — 223.255.255.255	2,097,152	256
D	224.0.0.0 — 239.255.255.255	N / A	N / A

E	240.0.0.0 — 255.255.255.255	N / A	N / A

The number of networks and hosts is shown as "theoretical," since in the actual Internet standards the numbers 0 and 255 in each octet are always reserved for special uses and are not available for use in normal host addresses.

Addresses in classes A, B, and C are used for normal host-to-host communications. Class D addresses are reserved for *multicasting*, that is, messages sent from one host to a group of hosts. Applications for multicasting include routing messages, network status messages, conference calls, and delivery of real-time audio and video data across the Internet. Class E addresses (with one exception) are reserved for future applications. (The exception is the address 255.255.255.255, which is a *broadcast* address. Broadcast addresses are discussed below.)

IP Subnet Addressing

Large organizations are usually assigned blocks of IP addresses that are allocated internally according to an administrative plan. To make IP addressing and routing easier inside a large organization network, it is possible to subdivide a large IP address space into a number of *subnets.* Subnets have the characteristics of networks for internal routing purposes, but the collection of subnets appears as one large network to hosts outside the organization.

Subnet addressing is achieved by taking the host number part of an IP address, and further dividing it into a subnet number and a host number. For example, many Class B networks are divided into (up to) 254 subnets, each of which can have up to 254 hosts each. (This is equivalent to having obtained assignment of 254 contiguous Class C networks, but is easier to administer, since they appear to be one large network to the outside world.)

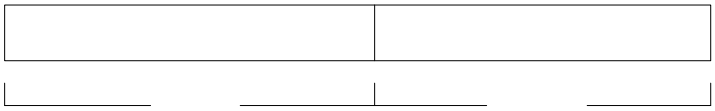


Figure B.1 : an unsubnetted Class B address

By using the third octet of the address as a subnet number, the organization's numbering authority can assign subnets to individual LANs that are connected by internal routers.

Figure B.2: a Class B address with 8-bit subnets

The value of subnets, besides ease of administration, is in simplification of network routing from outside the organization. Without subnets, routing information for each of the organization's internal LANs would have to be maintained by the Internet trunk routers. Using subnets, however, a single route for all the organization's subnets is the only one that needs to be known outside the organization.

When installing network software on a host in a subnetted environment, it is necessary to enter the appropriate *subnet mask* so that the host's address is properly interpreted. IP uses the subnet mask to determine which parts of the IP address should be treated as the network/subnet number and which as the host number. In a subnet mask, 1-bits are used for the network/subnet number, and 0-bits as the host number. Thus an unsubnetted Class B network would have the subnet mask 255.255.0.0 (upper 16 bits are network number, lower 16 bits are host number), while a Class B network using the third octet of the address as a subnet number would have the mask 255.255.255.0 (upper 24 bits are network/subnet number, lower 8 bits are host number). It is not necessary for the number of bits used for the subnet to be evenly divisible by 8. Organizations can declare larger or smaller subnets as necessary; however, 8-bit subnets are by far the most common and make dotted-decimal addresses easy to interpret. In the typical Class B subnetted scheme with a subnet mask of 255.255.255.0 the address 128.32.100.5 would be read as "Network 128.32, subnet 100, host 5."

Broadcast addresses

A host may often need to send a particular message to all the hosts on its local network, or all the hosts on a larger organization network. This is known as a *broadcast*. Uses for broadcasts include network status messages, routing messages, requests for a host's own IP address during rebooting, etc.

The basic broadcast destination address is 255.255.255.255, which means "all hosts on this local network". (Broadcast messages are not forwarded by routers, except for subnet broadcasts.) In a subnetted environment, the broadcast address 128.32.255.255 would mean "all hosts on all subnets of network 128.32." This broadcast would be carrier by internal routers to all subnet LANs on network 128.32. In practice, however, broadcasting is usually limited to a local network.

In earlier implementations, a host number of all 0-bits (e.g., 128.32.0.0) was used as the broadcast address. Though the standard has changed to a host number of all 1-bits (e.g., 128.32.255.255), most current implementations recognize the earlier form as well.

Future IP addressing issues

The IP 32-bit address model has proven to be very flexible and has served the Internet well, even during its recent explosive growth. However, it has become clear that for some purposes the current system will be inadequate in the future. One of the most pressing problems is the rapid increase in the number of individual networks in the Internet. Since IP routing requires that Internet trunk routers exchange nearly complete information about the gateway routers for each site, the burden of doing so may become unmanageable in the future. In addition, some IP address types — notably Class B addresses — are in danger of becoming scarce because of their finite availability. A number of solutions for these problems have been proposed, some of them requiring replacement of IP with another protocol at the Internetwork layer; some provide for extension of the 32-bit address space; still others propose changes to the IP routing model. Any change that is made will need to provide adequate backward compatibility for the very large number of hosts that cannot easily

adopt a new protocol implementation because of vendor or service provider issues.

While waiting for the implementation and deployment of IP v6 that sports a vastly enlarged addressing space, the most promising stopgap solution seems to rely on the so-called classless addressing (CIDR: Classless Inter-Domain Routing), as documented in RFC1517/1518/1519/1520. CIDR is based on the suppression of the concept of "network class" and, besides allowing network sizes more closely tailored to real needs, greatly simplifies the routing aggregating groups of old networks under "prefixes" of variable bit length. For example, IMA's prefix 202.75.0.0/18 includes four "classic" class C networks from 202.75.0.0 to 202.75.3.0; each route mentioning it replaces four routes in the old-style arrangement. This addressing space may be subnetted under the exclusive control of the local administration, without burdening the backbone core routers. With shorter prefixes, of course, the savings are even more impressive.

THE TCP/IP PROTOCOLS

Internet Protocol (IP)

The key feature of IP is its ability to route packets containing arbitrary higher-level protocol data throughout an internetwork that may consist of many different kinds of data-link and physical layers. This is done by *encapsulating* a transport-layer (i.e., TCP or UDP message) into an IP packet, adding sufficient header information to allow the packet to be correctly routed to a destination, and then preparing whatever framing information is necessary for the data-link layer to process the packet, and finally handing it off to the driver for the data-link layer implementation (e.g., Ethernet).

An important function of IP beyond message encapsulation is fragmentation and reassembly of packets, where the size of the encapsulated message is larger than that which can be handled by the underlying data-link layer network.

An IP packet consists of a header, the fields of which are shown below, and data, which is an arbitrary-length (up to 64K bytes) bit string consisting of a message prepared by a higher-level protocol.

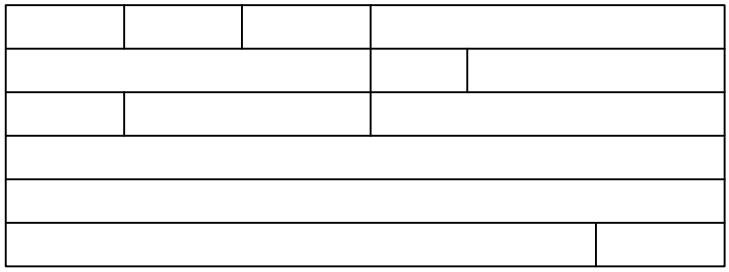


Figure B.3: IP packet header

The meaning of the header fields is shown below.

version	IP (protocol) version number
hlen	Header length in 32-bit words
tos	Type of service — usually unused
length	Length of entire packet, including header, in bytes
fragid, flags, fragoff	Fragment ID, fragment flags, and fragment offset — used for fragmentation and reassembly of large packets
ttl	Time-to-live — maximum lifetime of this packet, in seconds. Decremented by each router the packet passes through; effectively a maximum hop count, to prevent looping.
protocol	The protocol responsible for the encapsulated message, e.g., TCP, UDP, or others
checksum	A 16-bit checksum of the header only. Other error checking is performed by higher-level protocols on the encapsulated data field of the packet.
source	IP address of the source (originator) of the packet
destination	IP address of the ultimate destination of the packet
options	Various IP processing options, including source routing, route recording, time stamping, and security. Rarely used except for special purposes.
padding	Padding to bring the header to an even multiple of 32 bits

After IP receives a message from a transport-layer protocol, encapsulates it into an IP packet, and creates the IP packet header, IP determines which network interface will be used (possibly by consulting a routing table), then the packet is *framed* in the format appropriate for the data-link layer of that interface, and passed to a driver that arranges for submission of the packet onto the physical network.

In this process, the IP packet may be fragmented in order to comply with the maximum frame size of the data-link layer. While the maximum size of an IP packet is 64K bytes (because of the 16-bit field used to store the packet length), data-link layer protocols may allow only shorter frames; for example, the maximum transmission unit (MTU) of an Ethernet network is 1500 bytes (excluding headers).

In order to send a large IP packet through a network with smaller MTU's, IP will fragment the packet into two or more packets, and use the *fragid*, *flags*, and *fragoff* header fields to identify the size and order of the fragmented packet. The IP implementation receiving the packet holds the fragmented packets in a reassembly queue, and using the information in the header, will re-assemble the packets before presenting them to a higher-level protocol for processing.

Fragmentation and reassembly of packets is a relatively costly operation. IP uses various strategies to avoid fragmentation when sending packets across networks where the MTU is known in advance, and reducing packet size to a known minimum where the destination is many hops away.

Support Protocols for IP

Internet Control Message Protocol (ICMP)

ICMP is a support protocol for IP that is implemented by encapsulation into an IP packet. It is used for exchanging network control and informational messages among hosts and routers, and for diagnosis and debugging of network problems. Each ICMP packet contains a *type* and *code* field that specify particular queries, responses, or conditions. Some frequently used ICMP packet types are the following:

echo request, echo reply	The echo request is a packet with a small amount of data directed at a particular host, with a request to echo it (send it back); the echo reply contains the return data.
destination unreachable	Sent by a router to indicate the destination of an IP packet is not available (no route)
redirect	Sent by a router to indicate a better route to the destination host
source quench	Indicates network congestion or overflow; directs the source (sender of packets) to decrease the rate of transmission
time exceeded	Indicates that a packet's *ttl* has expired, usually because of a routing problem or loop

Address Resolution Protocol (ARP)

ARP is used to discover the mapping between IP addresses and hardware-based addresses of an underlying data-link layer network, such as Ethernet. IP addresses are assigned by a central numbering authority, and are implemented in software. However, some data-link network schemes, including Ethernet, token ring, and FDDI, use addresses that are embedded in network adapters. When IP sends a packet to another host on the same local network, it needs to construct a frame to be given to the lower-level network driver. Part of this frame is the hardware address of the destination IP host.

In order to obtain the hardware address of the destination host, the sending host looks the address up in its local *ARP cache* (table in memory of address mappings), and if it is not found, creates an ARP request packet, frames it with the proper data-link layer header, and broadcasts it to all hosts on the local

network. The ARP packet contains fields for the sender's IP and hardware address, and the recipient's IP and hardware address. (The recipient's hardware address, being unknown, is set to all zeros.)

When each host receives the broadcast, it matches its IP address with the destination address, and if it is the same, it fills in its hardware address in the recipient field, and sends the packet back to the sender address. The sender receives and decodes the packet and updates its ARP cache.

Reverse Address Resolution Protocol (RARP)

RARP is used almost exclusively for diskless workstations or intelligent terminals to discover their own IP address since they do not store it between operating sessions. In this case, a host will know its own hardware address (which is embedded in a ROM, for example), and needs to learn its IP address in order to receive a bootstrap image and engage in further IP protocol traffic.

Use of RARP requires setting up a RARP server that contains hardware address to IP address mappings. RARP packets are analogous to ARP packets — that is, the sending host will fill in the fields of the packet, frame it with the proper data-link layer header, and broadcast it on the local network.

Any RARP server can accept and decode the packet, fill in the requested IP address from its table, and return the packet to the sender.

IP ROUTING

Routing in IP is done on a per-network basis, that is, IP uses only the network number portion of an IP address to make routing decisions. This makes routing relatively simpler, since there is no need to keep a list of the location of all hosts in an internetwork (which would be practically impossible in the global Internet).

The basic element of IP routing is the routing table kept by each host, which is basically a list of network numbers and the IP address of a gateway (router) to which packets for that network should be sent. There is also usually a third field, a set of flags that indicate the status of the network and whether it is directly connected to the sending host.

For each IP packet to be sent, the sending host looks up the network number of the destination in the routing table, and determines whether the network number is that of the local, directly connected network. If so, the packet is sent directly to the destination (possibly using ARP to discover the hardware address of the destination on the local network). If the network number is not local, but there is an entry for it, the packet will be sent to the gateway listed for that network. Finally, if there is no entry matching the destination network number, the packet is discarded and a failure message is returned.

This basic routing algorithm is used by all IP hosts and routers. However, in practice, most hosts have a very small routing table, consisting of only three entries: an entry for itself (usually known as the "localhost" or "loopback" address; this is used for debugging and some software applications); an entry for its directly attached local network, for which the gateway field will be its own IP address — indicating that the packet should be delivered directly via the data-link layer; and an entry labeled "default," which means all other

packets that are non-local, for which the gateway field will be the IP address of the router that connects the local network to the Internet.

Routing Protocols

In the early days of the Internet, overall routing was managed by a set of core routers, known as the *mailbridges*, which contained routing table entries for every known connected network in the Internet. Every site network communicated directly with these core routers and propagated their routes throughout the Internet; some sites with limited capacity used the core routers as default routers. Because of the dramatic increase in the size of the connected Internet, it is no longer possible to maintain this method of routing.

The present Internet is divided into a collection of Autonomous Systems (AS) which are a set of routers that are under a single administration and exchange routing information among themselves using a specific routing protocol. The routing protocol shared by routers inside an AS is called an Interior Gateway Protocol; (IGP). Examples of IGPs are Routing Information Protocol (RIP) and Open Shortest Path First (OSPF).

Each AS has one or more exterior routers, which communicate with routers belonging to another AS. This involves the use of Exterior Gateway Protocols (EGP). Examples of EGPs are the original Exterior Gateway Protocol itself (EGP) and Border Gateway Protocol, version 4 (BGP 4).

Routing Information Protocol (RIP)

RIP is a simple routing protocol appropriate for a small network that is stable, homogeneous, and having a simple topology, such as a LAN router with two or more external WAN connections. RIP is a “distance vector” protocol, that is, it maintains a “cost” or “distance” metric for routes to distant networks.

While RIP supports requests for routing tables from remote routers, it is most commonly used as a system of updates whereby routers broadcast their routing tables in a form consisting of a network number and a distance metric. This distance metric is a hop count, that is, the number of intermediate routers through which a packet to that network must pass. Routers and hosts receiving RIP updates simply update their tables with this new information, and use the hop count to determine a “best” route to each network.

Open Shortest Path First (OSPF)

OSPF is suitable for large, complex autonomous systems that need to exchange routing information where network topology changes rapidly, or where there are multiple, different-cost paths to various locations. OSPF is relatively new to the Internet (deployed 1991) and is not in as wide use as RIP.

Routers using OSPF receive routing updates that include link-state and bandwidth information as well as distance vectors. This information is used to build internal graphs of internetwork topology that are used to generate shortest-path network routes.

OSPF also allows the definition of "routing areas" within an AS — in some ways the analog of IP subnets. This hides the details of complex routing inside parts of an AS.

Unlike RIP, OSPF requires the resources of a high-capacity dedicated router and therefore is usually only used when required by the complexity or criticality of a network.

Exterior Gateway Protocol (EGP)

EGP is a simple "reachability" protocol that allows AS routers to exchange information about which networks can be reached through their respective AS gateways. In EGP, there is a "core" AS that communicates directly with each other AS (which are known as "stubs"). Each stub AS supplies the core with information indicating what networks it handles traffic for, and the core redistributes this information to each other stub.

Border Gateway Protocol (BGP-4)

BGP is an extension to EGP that allows for AS's to be connected in other than a strict core-stub topology. The EGP "core" is redefined as a "Transit AS", that is, a network (or collection of networks) that permits transit traffic — connections where neither the source nor destination is on that AS. In addition, AS's can be stated to be either stubs or multi-homed; this allows for more complex reachability routing information.

The most important feature of BGP, besides additional robustness, is that it permits an AS to accept or reject transit traffic. This is used to implement so-called *policy-based routing*, where IP routing decisions are made on externally described policy factors such as type of traffic, as well as strictly on reachability and best-path factors.

TRANSPORT-LAYER PROTOCOLS

Transmission Control Protocol (TCP)

TCP is the most heavily used of the transport protocols in the TCP/IP suite. A TCP connection is a reliable, sequenced, bi-directional data stream between processes on two Internet hosts. This means that lost data will be retransmitted, the messages are guaranteed to arrive in the proper order, and communications can take place in either direction.

A TCP message is encapsulated inside an IP packet, and routed to its ultimate destination by IP, locally or through an arbitrary number of routers. The TCP message consists of a header, shown below, and data, which is a stream of bits prepared by (and meaningful to) an application-layer protocol (such as FTP or Telnet).

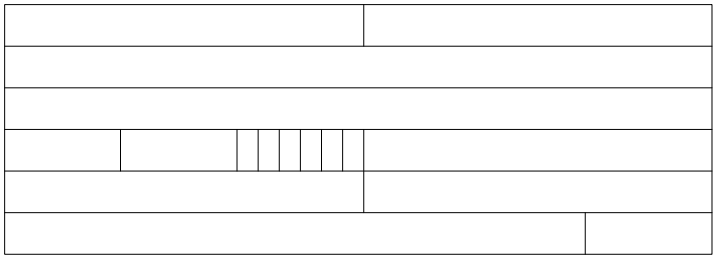


Figure B.4: TCP packet header

The meaning of the header fields is shown below.

source port	Port number of the source application. A list of port numbers used in TCP is in Figure B.5
destination port	Port number of the destination application
sequence number	Sequence number of this segment — used to assure correct ordering
acknowledgment number	Contains next segment number which is also used as acknowledgment of previous segments
data offset	Location of the data portion of packet (e.g., length of TCP header) in 32-bit words
URG bit	Notes "urgent data" is present
ACK bit	Notes Acknowledgment field is valid
PSH bit	Causes this data to be pushed through to application instead of buffered
RST bit	Causes connection to be reset
SYN bit	Causes sequence numbers to be resynchronized
FIN bit	Notes end of source data stream
window size	Size of flow control window
checksum	16-bit header and data checksum
urgent pointer	Location of "urgent data" inside packet
options	TCP options including maximum segment size, security, etc.
padding	Padding to next multiple of 32 bits.

The application on the originating host (called the client) and the application on the destination host (called the server) must establish a *connection* in order to

communicate using TCP. This is done by a somewhat complex initiation procedure that sets up the sequence numbers, window size, and other connection parameters. First, the client sends a message with the SYN bit set and includes a sequence number (this is generated by software using a real-time clock, to prevent sequence numbers from being reused too often).

The server accepts the SYN message, replies with a SYN message, includes its own sequence number, and sets the ACK bit as well. The client and server then negotiate other parameters of the connection, such as the window size (number of unacknowledged packets-in-flight that are permitted), and begin communicating according to whatever higher-level protocol data is being exchanged. During the connection, each side continues to increment the sequence number of their packets, and issues acknowledgments for packets it has received; in this way lost data will be recognized and re-transmitted after a time-out. For the purposes of flow control, either side may re-set the window size to more efficiently react to the speed and congestion of the network connection.

Finally, when one side proposes ending the connection, it sends a message with the FIN bit set; communication continues until the other side sends a FIN message as well.

TCP is probably the most complex protocol of the TCP/IP suite, because efficient network usage, high throughput, and reliability all depend on TCP's flow control and retransmission algorithms. For additional information on TCP specifics, see RFC-793.

Protocol	Port No.	Service
tcpmux	1	TCP port service multiplexer (RFC-1078)
rje (old)	5	Remote job entry (former)
echo	7	Echo (used for testing)
discard	9	Discard (used for testing)
systat	11	System status & users
daytime	13	Time of day
netstat	15	Network status
chargen	19	Character generator
ftp-data	20	File Transfer Protocol — data channel
ftp	21	File Transfer Protocol — control channel
telnet	23	TELNET protocol
smtp	25	Simple Mail Transfer Protocol
time	37	Time server
whois	43	NICname server
domain	53	Domain Name Service
gopher	70	Gopher information service
rje	77	Remote Job Entry
finger	79	Finger (user information) service
link	87	Terminal linking
supdup	95	SUPDUP (enhanced) Telnet protocol

hostnames	101	NIC hostname service (old)
x400	103	X.400 messaging
x400-snd	104	X.400 messaging
pop2	109	Post Office Protocol, Version 2
pop3	110	Post Office Protocol, Version 3
sunrpc	111	Sun Remote Procedure Call protocol
nntp	119	Network News Transport Protocol (Usenet)
ntp	123	Network Time Protocol
news	144	Network-extensible Window System
xdm	177	X Display Manager protocol
wais	210	Wide-Area Information Service
exec	512	Remote program execution (UNIX rexec)
login	513	Remote login (UNIX rlogin)
shell	514	Remote shell (UNIX rsh)
printer	515	Printer service (UNIX lpd)
courier	530	Courier remote procedure call protocol
uucp	540	UUCP over TCP

Figure B.5: TCP port assignments (partial list)

User Datagram Protocol (UDP)

UDP is a connectionless protocol that is useful when an application does not require 100% reliable delivery and sequencing of packets. This could be for one of two reasons — first, where the application performs error checking and sequencing itself, and arranges for re-transmission of lost data, and secondly, where integrity of every single packet of a voluminous stream is not needed. Examples of the former include the Network File System (NFS) where NFS itself employs numerous reliability and integrity checks; examples of the latter include broadcast-stream applications like voice or video distribution. Other services using UDP include Domain Name System (DNS) queries and replies, RIP routing updates, and services based on Sun's Remote Procedure Call (SunRPC) interface.

The UDP packet header is very simple compared with TCP. UDP also uses the abstraction of port numbers to identify various services. Since port numbers exist on a per-protocol basis, it is possible for UDP and TCP to have the same port number for the same or even a different service. As can be seen in Figure B.5, a number of UDP port numbers are the same as TCP port numbers; in this case, implementations of a protocol can choose whether to use TCP or UDP for the application. (For example, while DNS queries normally use UDP, most implementations can be configured to use TCP if requested by the user.)

The UDP header has only four parts, analogous to the same fields in TCP.

Figure B.6: UDP packet header

The meaning of the header fields is shown below.

source port	Port number of the source application. A list of port numbers used in UDP is in Figure B.7
destination port	Port number of the destination application
length	Length of the packet, including data
checksum	16-bit checksum of header and data

In practice, the UDP header source port field need not be filled in, since (as a connectionless protocol) there is no requirement that the destination's UDP server process reply to the source. Additionally, the checksum is optional and for the sake of further streamlining may not be decoded and computed by the destination.

Protocol	Port No.	Service
echo	7	Echo (used for testing)
discard	9	Discard (used for testing)
daytime	13	Network status
chargen	19	Character generator
time	37	Time server
name	42	NICname server
domain	53	Domain Name Service
tftp	69	Trivial File Transfer Protocol
sunrpc	111	Sun Remote Procedure Call protocol
snmp	161	Simple Network Monitoring Protocol
snmp-trap	162	SNMP Trap (event log)
biff	512	Mail arrival notification
who	513	Number of users and load average (UNIX)
syslog	514	System log messages
talk	517	Terminal-to-terminal chat (old)
ntalk	518	Terminal-to-terminal chat (new)
route	520	Routing Information Protocol (RIP)

new-rwho	550	Number of users/load (experimental)
rmonitor	560	Remote host monitoring (experimental)
monitor	561	Host monitoring (experimental)

Figure B.7: UDP port assignments (partial list)

APPLICATION PROTOCOLS

Telnet

The telnet protocol is the basic means of logging into a remote host over a TCP/IP network. The protocol is designed to be both robust enough to maintain a connection over a noisy transcontinental Internet link, while at the same time efficient enough not to slow down a user logging into a host on the same LAN. Telnet is a TCP-based protocol.

The telnet client is a program on the user's local system, often a personal computer or workstation. The telnet server is a program on a destination system that is able to accept login connections (normally a time-sharing system, but single-user personal computers may also have telnet servers). The telnet server listens for connection requests on TCP port 23. The session begins when a telnet client sends a request to open a connection.

The server accepts the connection request for processing, and the client and server negotiate telnet options (parameters for the connection). There are numerous telnet options, including whether the connection should be character-oriented (the normal case) or line-oriented; what characters should be used for special purposes like *interrupt* or *flush output*; carriage-return/line-feed mapping options; debugging information; what type of terminal is being emulated, etc. This negotiation is performed by exchanging a number of telnet protocol commands and assertions in the form WILL <option> / WONT <option> and DO <option> / DONT <option>. For example, a client may assert WILL FLUSH (flush output when an interrupt occurs), and the server may reply either DO FLUSH or DONT FLUSH indicating acceptance or rejection of that parameter.

At the end of the negotiation procedure, the server starts a login session for the client, and according to the parameters negotiated, the client and server programs pass characters back and forth, possibly interpreting or translating special characters or escape sequences. The protocol provides for re negotiation of options during the session if requested by either side.

When the session is ended, the client and server exchange closing messages and the connection is closed. The TCP implementation will detect the closing of the telnet session and terminate the underlying TCP connection.

File Transfer Protocol (FTP)

FTP is a method of copying files across TCP/IP networks. It is designed to interoperate among a very wide range of operating systems and file formats. FTP is mostly used between dissimilar hosts or host on different networks; on LANs, it is often simpler and more efficient to use a protocol such as *rcp* (UNIX

remote copy) or a file-sharing scheme (such as NFS) where network file transfers are transparent to users. FTP is also commonly used where the user does not have an account on the remote host, but certain files have been made available for public copying (anonymous FTP).

The FTP protocol and client-server model are somewhat complex, in that FTP sets up *two* TCP connections between the client and server. One connection carries commands and responses; the other carries the actual file data. The FTP protocol is interactive, in that the FTP user opens a connection to a remote server, set a number of options establishing the parameters of the connection, and then upload or download files or groups of files, and possibly navigate among directories on the server host.

FTP clients understand a number of user commands that are passed to the FTP server. These commands are usually typed to a command processor (in the case of DOS or UNIX) or may be buttons or fill-in fields in a graphical user interface such as Microsoft Windows. These commands are used to manage the connection with the remote host, supply required login and password information, move among directories and file systems; get lists of files and directories; determine the format of file presentation and transfer; upload and download files; determine if special processing of the files is required; rename, move, or delete remote files; and manage the local user interface.

FTP is implemented by submitting commands and protocol requests over the FTP control TCP connection; the server will reply over the same channel with protocol responses. Some commands/responses will result in data being transferred (in either direction) over the data TCP connection.

Each protocol response has an assigned numerical code, which is used by the client software. The verbal portion of the response code may or may not be seen directly by the user. The first digit of the response code indicates the family of the response, according to the following rules:

Leading digit	Meaning
1 _ _	Positive preliminary reply. The requested action is being initiated; expect another reply before proceeding with a new command.
2 _ _	Positive completion reply. The requested action has been successfully completed.
3 _ _	Positive intermediate reply. The command has been accepted, but the requested action is being held in abeyance, pending receipt of further information.
4 _ _	Transient negative completion reply. The command was not accepted and the requested action did not take place, but the error condition is temporary and the action may be requested again.
5 _ _	Permanent negative completion reply. The command was not accepted and the requested action did not take place. The user is discouraged from repeating the exact request (in the same sequence).

The second digit of the response code indicates the type of the response message:

Second digit	Meaning
_ 0 _	Syntax error
_ 1 _	Informational message
_ 2 _	Connection message
_ 3 _	Authentication / accounting message
_ 4 _	(reserved for future use)
_ 5 _	File system message

A complete list of FTP server response codes appears below.

110 Restart marker reply.
120 Service ready in nnn minutes.
125 Data connection already open; transfer starting.
150 File status okay; about to open data connection.

200 Command okay.
202 Command not implemented, superfluous at this site.
211 System status, or system help reply.
212 Directory status.
213 File status.
214 Help message.
215 NAME system type.
220 Service ready for new user.
221 Service closing control connection (user is logged out if appropriate).
225 Data connection open; no transfer in progress.
226 Closing data connection; requested file action successful.
227 Entering Passive Mode.
230 User logged in, proceed.
250 Requested file action okay, completed.
257 “PATHNAME” created.

331 User name okay, need password.
332 Need account for login.
350 Requested file action pending further information.

421 Service not available, closing control connection.
425 Can't open data connection.
426 Connection closed; transfer abo rted.
450 Requested file action not taken; e.g., file not present.
451 Requested action aborted: local error in processing.
452 Requested action not taken: insufficient storage space in system.

500 Syntax error, command unrecognized.
501 Syntax error in parameters or arguments.
502 Command not implemented.
503 Bad sequence of commands.
504 Command not implemented for that parameter.
530 Not logged in.
532 Need account for storing files.

550 Requested action not taken: file unavailable
551 Requested act ion aborted: page type unknown.
552 Requested file action aborted: exceeded storage allocation
553 Requested action not taken: file name not allowed.

Simple Mail Transfer Protocol (SMTP)

SMTP is the protocol used to transport electronic mail on the Internet. In addition to transporting mail between and among directly connected Internet sites, SMTP can also be used as a common mail backbone between organizations that use other types of mail systems, with the intermediate transport performed over Internet mail relays. **Internet Exchange** uses SMTP to send and receive messages over the Internet.

SMTP is closely associated with a specific message format, known as RFC822 format, which is the basic mail message type used in the Internet. Until the arrival of MIME (see Appendix A), RFC822 messages were the only standard message type carried by SMTP.

SMTP itself is a relatively simple "lock-step" protocol implemented using TCP. (Several non-TCP/IP implementations exist for SMTP, but are not in general use on the Internet.) Hosts that can receive SMTP messages run a SMTP server daemon that listens for connection requests on TCP port 25. Hosts wishing to send a SMTP message use a SMTP client program that opens a connection with an SMTP server on the destination host or a mail relay, and the client and server communicate in a series of protocol commands and responses. SMTP is called a "lock-step" protocol since every command has a finite number of protocol responses, and each side waits, synchronously, for the end of a protocol request or response before continuing.

When a user sends an Internet mail message (from a directly connected system), his mail program (user agent) will pass it to a mail transport program, which determines (by consulting the Domain Name System), which Internet host the message should be delivered to. The system's SMTP client then initiates a connection with the SMTP server on that host. (If the user is using a mail gateway such as **Internet Exchange**, there may be a number of intermediate steps before the message is given to an SMTP client for Internet delivery.)

After the connection is initiated and the client identifies its domain name (with the HELO command), it communicates the address of the sender of the message (with the MAIL command), and the addresses of the recipient of the messages (with the RCPT command). The SMTP server acknowledges each recipient address as either valid or invalid. When the list of recipients is complete, the client uses the DATA command to indicate that the message itself is ready to be sent. The server receives and acknowledges the message, and the client closes the connection. The server agrees to the close of connection and this is detected by the TCP implementation that terminates the connection. At this point, if the original message has further recipients elsewhere, the SMTP client will open a connection to another host and repeat the process until all recipients have been delivered to.

The SMTP command set is relatively small.

HELO <domain>	Client identifies itself to the server with its domain name
MAIL FROM:<reverse-path>	Identifies sender of the message
RCPT TO:<forward-path>	Identifies a recipient address
DATA	Message text follows, terminated by a line with a dot (“.”) by itself
RSET	Abort the current mail transaction and reset the connection
SEND FROM:<reverse-path>	Deliver message to a terminal (now considered an optional extension)
SOML FROM:<reverse-path>	Deliver message to terminal or mail it (now considered an optional extension)
SAML FROM:<reverse-path>	Deliver message to terminal and mail it (now considered an optional extension)
VRFY <string>	Confirm that <string> is a valid address
EXPN <string>	Expand (show the addresses in) a local mail alias or mailing list (now considered an optional extension)
HELP	Return a list of SMTP commands and meanings (now considered an optional extension)
NOOP	No operation; requires acknowledgment
QUIT	Request to close connection
TURN	Request for server and client to exchange roles so that client can receive mail queued on server host (now considered an optional extension)

The SMTP server response codes are similar in form to those used in FTP, but because of the relative simplicity of the protocol, are a smaller set.

211 System status, or system help reply
214 Help message
220 <domain> Service ready
221 <domain> Service closing transmission channel
250 Requested mail action okay, completed
251 User not local; will forward to <forward-path>

354 Start mail input; end with <CRLF>. <CRLF>

421 <domain> Service not available, closing transmission channel
450 Requested mail action not taken: mailbox unavailable
451 Requested action aborted: local error in processing
452 Requested action not taken: insufficient system storage

500 Syntax error, command unrecognized
501 Syntax error in parameters or arguments
502 Command not implemented
503 Bad sequence of commands
504 Command parameter not implemented
550 Requested action not taken: mailbox unavailable
551 User not local; please tr y <forward-path>
552 Requested mail action aborted: exceeded storage allocation
553 Requested action not taken: mailbox name not allowed
554 Transaction failed

Since the original implementation and deployment of SMTP, there have been a number of proposals to extend the SMTP command set and range of services, including adding the capability of handling messages that are not 7-bit ASCII text. The standards document RFC-1651, “SMTP Service Extensions” (February 1993), defines a method of extending SMTP by registering additional capabilities with the central Internet naming authority, and listing them as part of a simple extensions negotiation at the beginning of an SMTP session. This extended SMTP protocol is called ESMTP.

APPENDIX E
INTERNET INTRODUCTION/REFERENCES/GLOSSARY

FYI ON QUESTIONS AND ANSWERS - ANSWERS TO COMMONLY ASKED "NEW INTERNET USER" QUESTIONS

by April Marine, Joyce Reynolds, and Gary Malkin
(Published as RFC-1594, FYI-4, March 1994; specially edited for inclusion)

Abstract

This FYI RFC is one of two FYI's called, "Questions and Answers" (Q/A), produced by the User Services Working Group of the Internet Engineering Task Force (IETF). The goal is to document the most commonly asked questions and answers in the Internet.

1. Introduction

New users joining the Internet community have the same questions as did everyone else who has ever joined. Our quest is to provide the Internet community with up to date, basic Internet knowledge and experience.
Future updates of this memo will be produced as User Services members become aware of additional questions that should be included, and of deficiencies or inaccuracies that should be amended in this document. Although the RFC number of this document will change with each update, it will always have the designation of FYI 4. An additional FYI Q/A, FYI 7, is published that deals with intermediate and advanced Q/A topics [11].

2. Acknowledgments

The following people deserve thanks for their help and contributions to this FYI Q/A: Matti Aarnio (FUNET), Susan Calcari (InterNIC), Corinne Carroll (BBN), Vint Cerf (MCI), Peter Deutsch (Bunyip), Alan Emtage (Bunyip), John Klensin (UNU), Thomas Lenggenhager (Switch), Doug Mildram (Xylogics), Tracy LaQuey Parker (Cisco), Craig Partridge (BBN), Jon Postel (ISI), Matt Power (MIT), Karen Roubicek (BBN), Patricia Smith (Merit), Gene Spafford (Purdue), and Carol Ward (Sterling Software/NASA NAIC).

3. Questions About the Internet

3.1 What is the Internet?

The Internet is a collection of thousands of networks linked by a common set of technical protocols that make it possible for users of any one of the networks to communicate with or use the services located on any of the other networks. These protocols are referred to as TCP/IP or the TCP/IP protocol suite. The Internet started with the ARPANET, but now includes such networks as the National Science Foundation Network (NSFNET), the Australian Academic and Research Network (AARNet), the NASA Science Internet (NSI), the Swiss Academic and Research Network (SWITCH), and about 10,000 other large and small, commercial and research, networks.

There are other major wide area networks that are not based on the TCP/IP protocols and are thus often not considered part of the Internet. However, it is possible to communicate between them and the Internet by electronic mail because of mail gateways that act as "translators" between the different network protocols involved.

Note: You will often see "internet" with a small "i". This could refer to any network built based on TCP/IP, or might refer to networks using other protocol families that are composites built of smaller networks.
See FYI 20 (RFC 1462), "FYI on 'What is the Internet?'" for a lengthier description of the Internet [13].

3.2 I just got on the Internet. What can I do now?

You now have access to all the resources you are authorized to use on your own Internet host, on any other Internet host on which you have an account, and on any other Internet host that offers publicly accessible information. The Internet gives you the ability to move information between these hosts by file transfers. Once you are logged into one host, you can use the Internet to open a connection to another, login, and use its services interactively (this is known as remote login or "TELNETing"). In addition, you can send electronic mail to users at any Internet site and to users on many non-Internet sites that are accessible by electronic mail.

There are various other services you can use. For example, some hosts provide access to specialized databases or to archives of information. The Internet Resource Guide provides information regarding some of these sites. The Internet Resource Guide lists facilities on the Internet that are available to users. Such facilities include supercomputer centers, library catalogs and specialized data collections. The guide is maintained by the Directory Services portion of the InterNIC and is available online in a number of ways. It is available for anonymous FTP from the host ds.internic.net in the resource-guide directory. It is also readable by the InterNIC gopher (gopher internic.net). For more information, contact admin@ds.internic.net or call the InterNIC at (800) 444-4345 or (908) 668-6587.

Today the trend for Internet information services is to strive to present the users with a friendly interface to a variety of services. The goal is to reduce the traditional needs for a user to know the source host of a service and the different command interfaces for different types of services. The Internet Gopher (discussed more in the "Questions about Internet Services" section) is one such service to which you have access when you join the Internet.

3.3 How do I find out if a site has a computer on the Internet?

Frankly, it's almost impossible to find out if a site has a computer on the Internet by querying some Internet service itself. The most reliable way is to ask someone at the site you are interested in contacting.

It is sometimes possible to find whether or not a site has been assigned an IP network number, which is a prerequisite for connecting an IP network to the Internet (which is only one type of Internet access). To do so, query the WHOIS database, maintained by the Registration Services portion of the InterNIC. You have several options about how to do such a query. The most common currently are to TELNET to the host rs.internic.net and invoke one of the search interfaces provided, or to run a WHOIS client locally on your machine and use it to make a query across the network.

The RIPE Network Coordination Center (RIPE NCC) also maintains a large database of sites to whom they have assigned IP network numbers. You can query it by TELNETing to info.ripe.net and stepping through the interactive interface they provide.

3.4 How do I get a list of all the hosts on the Internet?

You really don't want that. The list includes more than 1.5 million hosts. Almost all of them require that you have access permission to actually use them. You may really want to know which of these hosts provide services to the Internet community. Investigate using some of the network resource discovery tools, such as gopher, to gain easier access to Internet information.

4. Questions About TCP/IP

4.1 What is TCP/IP?

TCP/IP (Transmission Control Protocol/Internet Protocol) [4,5,6] is the common name for a family of over 100 data-communications protocols used to organize computers and data-communications equipment into computer networks. TCP/IP was developed to interconnect hosts on ARPANET, PRNET (packet radio), and SATNET (packet satellite). All three of these networks have since been retired; but TCP/IP lives on. It is currently used on a large international network of networks called the Internet, whose members include universities, other research institutions, government facilities, and many corporations. TCP/IP is also sometimes used for other networks, particularly local area networks that tie together numerous different kinds of computers or tie together engineering workstations.

4.2 What are the other well-known standard protocols in the TCP/IP family?

Other than TCP and IP, the three main protocols in the TCP/IP suite are the Simple Mail Transfer Protocol (SMTP) [8], the File Transfer Protocol (FTP) [3], and the TELNET Protocol [9]. There are many other protocols in use on the Internet. The Internet Architecture Board (IAB) regularly publishes an RFC [2] that describes the state of standardization of the various Internet protocols. This document is the best guide to the current status of Internet protocols and their recommended usage.

5. Questions About the Domain Name System

5.1 What is the Domain Name System?

The Domain Name System (DNS) is a hierarchical, distributed method of organizing the name space of the Internet. The DNS administratively groups hosts into a hierarchy of authority that allows addressing and other information to be widely distributed and maintained. A big advantage to the DNS is that using it eliminates dependence on a centrally-maintained file that maps host names to addresses.

5.2 What is a Fully Qualified Domain Name?

A Fully Qualified Domain Name (FQDN) is a domain name that includes all higher level domains relevant to the entity named. If you think of the DNS as a tree-structure with each node having its own label, a Fully Qualified Domain Name for a specific node would be its label followed by the labels of all the other nodes between it and the root of the tree. For example, for a host, a FQDN would include the string that identifies the particular host, plus all domains of which the host is a part up to and including the top-level domain (the root domain is always null). For example, atlas.arc.nasa.gov is a Fully Qualified Domain Name for the host at 128.102.128.50. In addition, arc.nasa.gov is the FQDN for the Ames Research Center (ARC) domain under nasa.gov.

6. Questions About Internet Documentation

6.1 What is an RFC?

The Request for Comments documents (RFCs) are working notes of the Internet research and development community. A document in this series may be on essentially any topic related to computer communication, and may be anything from a meeting report to the specification of a standard. Submissions for Requests for Comments may be sent to the RFC Editor (RFC-EDITOR@ISI.EDU). The RFC Editor is Jon Postel.

Most RFCs are the descriptions of network protocols or services, often giving detailed procedures and formats for their implementation. Other RFCs report on the results of policy studies or summarize the work of technical committees or workshops. All RFCs are considered public domain unless explicitly marked otherwise.

While RFCs are not refereed publications, they do receive technical review from either the task forces, individual technical experts, or the RFC Editor, as appropriate. Currently, most standards are published as RFCs, but not all RFCs specify standards.

Anyone can submit a document for publication as an RFC. Submissions must be made via electronic mail to the RFC Editor. Please consult RFC 1543, “Instructions to RFC Authors” [10], for further information. RFCs are accessible online in public access files, and a short message is sent to a notification distribution list indicating the availability of the memo. Requests to be added to this distribution list should be sent to RFC-REQUEST@NIC.DDN.MIL.

The online files are copied by interested people and printed or displayed at their sites on their equipment. (An RFC may also be returned via electronic mail in response to an electronic mail query.) This means that the format of the online files must meet the constraints of a wide variety of printing and display equipment.

Once a document is assigned an RFC number and published, that RFC is never revised or re-issued with the same number. There is never a question of having the most recent version of a particular RFC. However, a protocol (such as File Transfer Protocol (FTP)) may be improved and re-documented many times in several different RFCs. It is important to verify that you have the most recent RFC on a particular protocol. The "Internet Official Protocol Standards" [2] memo is the reference for determining the correct RFC to refer to for the current specification of each protocol.

6.2 How do I obtain RFCs?

RFCs are available online at several repositories around the world. For a list of repositories and instructions about how to obtain RFCs from each of the major U.S. ones, send a message to rfc-info@isi.edu. As the text of the message, type "help: ways_to_get_rfcs" (without the quotes).

An example of obtaining RFCs online follows.

RFCs can be obtained via FTP from ds.internic.net with the pathname rfc/rfcNNNN.txt (where "NNNN" refers to the number of the RFC). Login using FTP, username "anonymous" and your email address as password. The Directory Services portion of the InterNIC also makes RFCs available via electronic mail, WAIS, and gopher.

To obtain RFCs via electronic mail, send a mail message with the word "help" alone in the message body to mailserv@ds.internic.net; a help file will be returned to you.

6.3 How do I obtain a list of RFCs?

Several sites make an index of RFCs available. These sites are indicated in the ways_to_get_rfcs file mentioned above and in the next question.

6.4 What is the RFC-INFO service?

The Information Sciences Institute, University of Southern California (ISI) has a service called RFC-INFO. Even though this is a service, rather than a document, we'll discuss it in this section because it is so closely tied to RFC information.

RFC-INFO is an email based service to help in locating and retrieval of RFCs, FYIs, STDs, and IMRs. Users can ask for "lists" of all RFCs and FYIs having certain attributes ("filters") such as their ID, keywords, title, author, issuing organization, and date. Once an RFC is uniquely identified (e.g., by its RFC number) it may also be retrieved.

To use the service, send email to: RFC-INFO@ISI.EDU with your requests as the text of the message. Feel free to put anything in the SUBJECT, the system ignores it. All input is case independent. Report problems to: RFC-MANAGER@ISI.EDU.

To get started, you may send a message “HELP” (without the quotes) to RFC-INFO@ISI.EDU.

6.5 Which RFCs are Standards?

See “Internet Official Protocol Standards” (currently RFC 1540) [2]. This RFC documents the status of each RFC on the Internet standards track, as well as the status of RFCs of other types. It is updated periodically; make sure you are referring to the most recent version. In addition, the RFC Index maintained at the ds.internic.net repository notes the status of each RFC listed.

6.6 What is an FYI?

FYI stands for For Your Information. FYIs are a subset of the RFC series of online documents.

FYI 1 states, “The FYI series of notes is designed to provide Internet users with a central repository of information about any topics which relate to the Internet. FYI topics may range from historical memos on 'Why it was done this way' to answers to commonly asked operational questions. The FYIs are intended for a wide audience. Some FYIs will cater to beginners, while others will discuss more advanced topics.”

In general, then, FYI documents tend to be more information oriented, while RFCs are usually (but not always) more technically oriented.

FYI documents are assigned both an FYI number and an RFC number. As RFCs, if an FYI is ever updated, it is issued again with a new RFC number; however, its FYI number remains unchanged. This can be a little confusing at first, but the aim is to help users identify which FYIs are about which topics. For example, FYI 4 will always be FYI 4, even though it may be updated several times and during that process receive different RFC numbers. Thus, you need only to remember the FYI number to find the proper document. Of course, remembering titles often works as well.

FYIs can be obtained in the same way RFCs can and from the same repositories. In general, their pathnames are fyi/fyiNN.txt or fyi/fyiNN.ps, where NN is the number of the FYI without leading zeroes.

6.7 What is an STD?

The newest subseries of RFCs are the STDs (Standards). RFC 1311 [12], which introduces this subseries, states that the intent of STDs is to identify clearly those RFCs that document Internet standards. An STD number will be assigned only to those specifications that have completed the full process of standardization in the Internet. Existing Internet standards have been assigned STD numbers; a list of them can be found both in RFC 1311 and in the, “Internet Official Protocol Standards” RFC.

Like FYIs, once a standard has been assigned an STD number, that number will not change, even if the standard is reworked and re- specified and later issued with a new RFC number.

It is important to differentiate between a “standard” and “document”. Different RFC documents will always have different RFC numbers. However, sometimes the complete specification for a standard will be contained in more than one RFC document. When this happens, each of the RFC documents that

is part of the specification for that standard will carry the same STD number. For example, the Domain Name System (DNS) is specified by the combination of RFC 1034 and RFC 1035; therefore, both of those RFCs are labeled STD 13.

6.8 What is the Internet Monthly Report?

The Internet Monthly Report (IMR) communicates online to the Internet community the accomplishments, milestones reached, or problems discovered by the participating organizations. Many organizations involved in the Internet provide monthly updates of their activities for inclusion in this report. The IMR is for Internet information purposes only.

You can receive the report online by joining the mailing list that distributes the report. Requests to be added or deleted from the Internet Monthly Report list should be sent to " imr- request@isi.edu".

In addition, back issues of the Report are available for anonymous FTP from the host ftp.isi.edu in the in-notes/imr directory, with the file names in the form imryymm.txt, where yy is the last two digits of the year and mm two digits for the month. For example, the July 1992 Report is in the file imr9207.txt.

6.9 What is an Internet Draft? Are there any guidelines available for writing one?

Internet Drafts (I-Ds) are the current working documents of the IETF. Internet Drafts are generally in the format of an RFC with some key differences:

- The Internet Drafts are not RFCs and are not a numbered document series.
- The words INTERNET-DRAFT appear in place of RFC XXXX in the upper left-hand corner.
- The document does not refer to itself as an RFC or as a Draft RFC.
- An Internet Draft does not state nor imply that it is a proposed standard. To do so conflicts with the role of the IAB, the RFC Editor, and the Internet Engineering Steering Group (IESG).

An Internet Drafts directory has been installed to make draft documents available for review and comment by the IETF members. These draft documents that will ultimately be submitted to the IAB and the RFC Editor to be considered for publishing as RFCs. The Internet Drafts Directories are maintained on several Internet sites. There are several "shadow" machines which contain the IETF and Internet Drafts Directories. They are:

West Coast (US)	ftp.isi.edu (128.9.0.32)
East Coast (US)	ds.internic.net (198.49.45.10)
Europe	nic.nordu.net (192.36.148.17)
Pacific Rim	munnari.oz.au (128.250.1.21)

To access these directories, use anonymous FTP. Login with username "anonymous" and your email address as password (or "guest" if that fails). Once logged in, change to the desired directory with "cd internet-drafts". Internet Draft files can then be retrieved. Once logged in, if you change to the directory "ietf", you can retrieve a file called "1id-guidelines.txt", which explains how to write and submit an Internet Draft.

6.10 How do I obtain OSI Standards documents?

OSI Standards documents are NOT available from the Internet via anonymous FTP due to copyright restrictions. These are available from:

Omnicom Information Service
501 Church Street NE, Suite 304
Vienna, Virginia 22180
USA
Telephone: (800) 666-4266 or (703) 281-1135
Fax: (703) 281-1505

American National Standards Institute
11 West 42nd Street
New York, NewYork 10036
USA
Telephone: (212) 642-4900

However, the GOSIP specification which covers the use of OSI protocols within the U.S. Government is available from the National Institute of Standards and Technology (NIST). The final text of GOSIP Version 2 is now available from both sites.

Online sources:

Available through anonymous FTP from osi.ncsl.nist.gov (129.6.48.100) as:
./pub/gosip/gosip_v2.txt -- ascii
./pub/gosip/gosip_v2.txt.Z -- ascii compressed
./pub/gosip/gosip_v2.ps -- PostScript
./pub/gosip/gosip_v2.ps.Z -- PostScript compressed

Hardcopy source:

Standards Processing Coordinator (ADP)
National Institute of Standards and Technology
Technology Building, Room B-64
Gaithersburg, Maryland 20899
Telephone: (301) 975-2816

7. Questions about Internet Organizations and Contacts

7.1 What is the IAB?

The Internet Architecture Board (IAB) is concerned with technical and policy issues involving the evolution of the Internet architecture [7]. IAB members are deeply committed to making the Internet function effectively and evolve to meet a large scale, high speed future. The chairman serves a term of two years and is elected by the members of the IAB. The IAB focuses on the TCP/IP protocol suite, and extensions to the Internet system to support multiple protocol suites.

The IAB performs the following functions:

1) Reviews Internet Standards,
2) Manages the RFC publication process,
3) Reviews the operation of the IETF and IRTF,
4) Performs strategic planning for the Internet, identifying long-range problems and opportunities,
5) Acts as an international technical policy liaison and representative for the Internet community, and

6) Resolves technical issues which cannot be treated within the IETF or IRTF frameworks.

The IAB has two principal subsidiary task forces:

1) Internet Engineering Task Force (IETF)
2) Internet Research Task Force (IRTF)

Each of these Task Forces is led by a chairman and guided by a Steering Group which reports to the IAB through its chairman. For the most part, a collection of Research or Working Groups carries out the work program of each Task Force.

All decisions of the IAB are made public. The principal vehicle by which IAB decisions are propagated to the parties interested in the Internet and its TCP/IP protocol suite is the Request for Comments (RFC) note series and the Internet Monthly Report.

7.2 What is the IETF?

The Internet has grown to encompass a large number of widely geographically dispersed networks in academic and research communities. It now provides an infrastructure for a broad community with various interests. Moreover, the family of Internet protocols and system components has moved from experimental to commercial development. To help coordinate the operation, management and evolution of the Internet, the IAB established the Internet Engineering Task Force (IETF).

The IETF is a large open community of network designers, operators, vendors, and researchers concerned with the Internet and the Internet protocol suite. The activity is performed in a number of working groups organized around a set of several technical areas, each working group has a chair, and each area is managed by a technical area director. The IETF overall is managed by its chair and the Internet Engineering Steering Group (IESG), which is made up of the area directors.

The IAB has delegated to the IESG the general responsibility for the resolution of short- and mid-range protocol and architectural issues required to make the Internet function effectively, and the development of Internet standards.

7.3 What is the IRTF?

To promote research in networking and the development of new technology, the IAB established the Internet Research Task Force (IRTF). The IRTF is a set of research groups, generally with an Internet focus. The work of the IRTF is governed by its Internet Research Steering Group (IRSG).

In the area of network protocols, the distinction between research and engineering is not always clear, so there will sometimes be overlap between activities of the IETF and the IRTF. There is, in fact, considerable overlap in membership between the two groups. This overlap is regarded as vital for cross-fertilization and technology transfer.

7.4 What is the Internet Society?

The Internet Society is a relatively new, professional, non-profit organization with the general goal of fostering the well-being and continued interest in, and evolution and use of the Internet. The Society (often abbreviated ISOC) is integrating the IAB, IETF, and IRTF functions into its operation.

The following goals of the Society are taken from its charter:

A. To facilitate and support the technical evolution of the Internet as a research and education infrastructure, and to stimulate the involvement of the scientific community, industry, government and others in the evolution of the Internet;
B. To educate the scientific community, industry and the public at large concerning the technology, use and application of the Internet;
C. To promote educational applications of Internet technology for the benefit of government, colleges and universities, industry, and the public at large;
D. To provide a forum for exploration of new Internet applications, and to stimulate collaboration among organizations in their operational use of the global Internet.

More information about the Internet Society is available for anonymous FTP from the host: isoc.org in the directory: isoc. Information is also available via the ISOC gopher, accessible via "gopher isoc.org" if you are running a gopher client.

7.5 What is the IANA?

The task of coordinating the assignment of values to the parameters of protocols is delegated by the Internet Architecture Board (IAB) to the Internet Assigned Numbers Authority (IANA). These protocol parameters include op-codes, type fields, terminal types, system names, object identifiers, and so on. The "Assigned Numbers" Request for Comments (RFC) [1] documents the currently assigned values from several series of numbers used in network protocol implementations. Internet addresses and Autonomous System numbers are assigned by the Registration Services portion of the InterNIC. The IANA is located at USC/Information Sciences Institute.

Current types of assignments listed in Assigned Numbers and maintained by the IANA are:

Address Resolution Protocol Parameters
BOOTP Parameters and BOOTP Extension Codes
Character Sets
Domain System Parameters
Encoding Header Field Keywords
ESMTP Mail Keywords
Ethernet Multicast Addresses
Ethernet Numbers of Interest
Ethernet Vendor Address Components
IANA Ethernet Address Block
ICMP Type Numbers
IEEE 802 Numbers of Interest
Internet Protocol Numbers
Internet Version Numbers

IP Option Numbers
IP Time to Live Parameter
IP TOS Parameters
Internet Multicast Addresses
Inverse Address Resolution Protocol
Machine Names
Mail Encryption Types
Mail System Names
Mail Transmission Types
MILNET X.25 Address Mappings
MILNET Logical Addresses
MILNET Link Numbers
MIME Types
MIME / X.400 Mapping Tables
Network Management Parameters
Novell Numbers
Operating System Names
OSPF Authentication Codes
Point-to-Point Protocol Field Assignments
Protocol Numbers
Protocol and Service Names
Protocol / Type Field Assignments
Public Data Network Numbers
Reverse Address Resolution Protocol
Operation Codes
SUN RPC Numbers
TCP Option Numbers
TCP Alternate Checksum Numbers
TELNET Options
Terminal Type Names
Version Numbers
Well Known and
Registered Port Numbers
X.25 Type Numbers
XNS Protocol Types

For more information on number assignments, contact: IANA@ISI.EDU.

7.6 What is a NIC? What is a NOC?

"NIC" stands for Network Information Center. It is an organization which provides network users with information about services provided by the network.

"NOC" stands for Network Operations Center. It is an organization that is responsible for maintaining a network.

For many networks, especially smaller, local networks, the functions of the NIC and NOC are combined. For larger networks, such as mid-level and backbone networks, the NIC and NOC organizations are separate, yet they do need to interact to fully perform their functions.

7.7 What is the InterNIC?

The InterNIC is a five year project partially supported by the National Science Foundation to provide network information services to the networking

community. The InterNIC began operations in April of 1993 and is a collaborative project of three organizations: General Atomics provides Information Services from their location in San Diego, CA; AT&T provides Directory and Database Services from South Plainsfield, NJ; and Network Solutions, Inc. provides Registration Services from their headquarters in Herndon, VA. Services are provided via the network electronically, and by telephone, FAX, and hardcopy documentation.

General Atomics offers Information Services acting as the "NIC of first and last resort" by providing a Reference Desk for new and experienced users, and midlevel and campus NICs. The InterNIC Reference Desk offers introductory materials and pointers to network resources and tools.

AT&T services include the Directory of Directories, Directory Services, and Database Services to store data available to all Internet users.

Network Solutions, Inc. (NSI) provides Internet registration services including IP address allocation, domain registration, and Autonomous System Number assignment. NSI also tracks points of contact for networks and domain servers and provides online and telephone support for questions related to IP address or domain name registration.

All three portions of the InterNIC can be reached by calling (800) 444-4345 or by sending a message to info@internic.net. Callers from outside the U.S. can telephone +1 (619) 445-4600. Extensive online information is available at host is.internic.net, accessible via gopher or TELNET.

7.8 What is the DDN NIC (nic.ddn.mil)?

The DDN NIC is the Defense Data Network NIC. Until the formation of the InterNIC, the DDN NIC had been responsible for many services to the whole Internet, especially for registration services. Now the DDN NIC focuses on serving its primary constituency of MILNET users. Its host is nic.ddn.mil; the address hostmaster@nic.ddn.mil may still be in older Internet registration documentation. The DDN NIC maintains close ties to the newer InterNIC.

7.9 What is the IR?

The Internet Registry (IR) is the organization that is responsible for assigning identifiers, such as IP network numbers and autonomous system numbers, to networks. The IR also gathers and registers such assigned information. The IR delegates some number assignment authority to regional registries (such as NCC@RIPE.NET and APNIC-STAFF@APNIC.NET). However, it will continue to gather data regarding such assignments. At present, the Registration Services portion of the InterNIC at Network Solutions, Inc., serves as the IR.

8. Questions About Services

8.1 How do I find someone's electronic mail address?

There are a number of directories on the Internet; however, all of them are far from complete. Many people can be found, however, via the InterNIC WHOIS services, or KNOWBOT. Generally, it is still necessary to ask the person for his or her email address.

8.2 How do I use the WHOIS program at the InterNIC Registration Services?

There are several ways to search the WHOIS database. You can TELNET to the InterNIC registration host, rs.internic.net. There is no need to login. Type "whois" to call up the information retrieval program, or choose one of the other options presented to you. Help is available for each option. You can also run a client of the WHOIS server and point it at any whois database you'd like to search. Pointing a client at the whois server ds.internic.net will enable you to query the databases at three hosts: ds.internic.net, rs.internic.net, and nic.ddn.mil.

For more information, contact the InterNIC at (800) 444-4345 or the registration services group at (703) 742-4777.

8.3 How do I use the Knowbot Information Service?

The Knowbot Information Service is a white pages "meta-service" that provides a uniform interface to heterogeneous white pages services in the Internet. Using the Knowbot Information Service, you can form a single query that can search for white pages information from the NIC WHOIS service, the PSI White Pages Pilot Project, and MCI Mail, among others, and have the responses displayed in a single, uniform format.
Currently, the Knowbot Information Service can be accessed through TELNET to port 185 on hosts cnri.reston.va.us and sol.bucknell.edu. From a UNIX host, use "telnet cnri.reston.va.us 185". There is also an electronic mail interface available by sending mail to netaddress at either cnri.reston.va.us or sol.bucknell.edu.

The commands "help" and "man" summarize the command interface. Simply entering a user name at the prompt searches a default list of Internet directory services for the requested information. Organization and country information can be included through the syntax: "userid@organization.country". For example, the queries "droms@bucknell" and "kille@ucl.gb" are both valid. Note that these are not Domain Names, but rather a syntax to specify an organization and a country for the search.

8.4 What is the White Pages at PSI?

Performance Systems International, Inc. (PSI), sponsors a White Pages Project that collects personnel information from member organizations into a database and provides online access to that data. This effort is based on the OSI X.500 Directory standard.

To access the data, TELNET to WP.PSI.COM and login as "fred" (no password is necessary). You may now look up information on participating organizations. The program provides help on usage. For example, typing "help" will show you a list of commands, "manual" will give detailed documentation, and "whois" will provide information regarding how to find references to people. For a list of the organizations that are participating in the pilot project by providing information regarding their members, type "whois -org *".

Access to the White Pages data is also possible via programs that act as X.500 Directory User Agent (DUA) clients.

For more information, send a message to WP-INFO@PSI.COM.

8.5 What is USENET? What is Netnews?

USENET is the formal name, and Netnews a common informal name, for a distributed computer information service that some hosts on the Internet use. USENET handles only news and not mail. USENET uses a variety of underlying networks for transport, including parts of the Internet, BITNET, and others. Netnews can be a valuable tool to economically transport traffic that would otherwise be sent via mail. USENET has no central administration.

8.6 How do I get a Netnews feed?

To get a Netnews feed, you must acquire the server software, which is available for some computers at no cost from some anonymous FTP sites across the Internet, and you must find an existing USENET site that is willing to support a connection to your computer. In many cases, this "connection" merely represents additional traffic over existing Internet access channels.

One well-known anonymous FTP archive site for software and information regarding USENET is ftp.uu.net. There is a "news" directory which contains many software distribution and information sub-directories.

It is recommended that new users subscribe to and read news.announce.newusers since it will help to become oriented to USENET and the Internet.

8.7 What is a newsgroup?

A newsgroup is a bulletin board which readers interested in that newsgroup's particular topic can read and respond to messages posted by other readers. Generally, there will be a few "threads" of discussion going on at the same time, but they all share some common theme. There are approximately 5000 newsgroups, and there are more being added all the time.

There are two types of newsgroups: moderated and unmoderated. A moderated newsgroup does not allow individuals to post directly to the newsgroup. Rather, the postings go to the newsgroup's moderator who determines whether or not to pass the posting to the entire group. An unmoderated newsgroup allows a reader to post directly to the other readers.

8.8 How do I subscribe to a newsgroup?

You don't subscribe to a newsgroup. Either you get it on your machine or you don't. If there's one you want, all you can do is ask the systems administrator to try to get it for you.

8.9 What is anonymous FTP?

Anonymous FTP is a conventional way of allowing you to sign on to a computer on the Internet and copy specified public files from it [3]. Some sites offer anonymous FTP to distribute software and various kinds of information. You use it like any FTP, but the username is "anonymous". Many systems will request that the password you choose is your email address. If this fails, the generic password is usually "guest".

8.10 What is "archie"?

The archie system was created to automatically track anonymous FTP archive sites, and this is still its primary function. The system currently makes available the names and locations of some 2,100,000 files at some 1,000 archive sites.

Archie's User Access component allows you to search the "files" database for these filenames. When matches are found, you are presented with the appropriate archive site name, IP address, the location within the archive, and other useful information.

You can also use archie to "browse" through a site's complete listing in search of information of interest, or obtain a complete list of the archive sites known to that server.

The archie server also offers a "package descriptions" (or "whatis") database. This is a collection of names and descriptions gathered from a variety of sources and can be used to identify files located throughout the Internet, as well as other useful information. Files identified in the whatis database can then be found by searching the files database as described above.

8.11 How do I connect to archie?

You can connect to archie in a variety of ways. There is a conventional TELNET interface, an electronic mail interface, and a variety of client programs available. The use of a client is strongly encouraged. There are currently 22 archie servers located throughout the world.

To try the TELNET interface to archie you can TELNET to one of the 22 archie servers (preferably the one nearest you, and during non-peak hours). Log in as "archie" (no password is required). Type "help" to get you started.

Here is a list of archie servers as of the date this was written:

archie.au*	139.130.4.6	Australia
archie.edvz.uni-linz.ac.at*	140.78.3.8	Austria
archie.univie.ac.at*	131.130.1.23	Austria
archie.uqam.ca*	132.208.250.10	Canada
archie.funet.fi	128.214.6.100	Finland
archie.th-darmstadt.de*	130.83.22.60	Germany
archie.ac.il*	132.65.6.15	Israel
archie.unipi.it*	131.114.21.10	Italy
archie.wide.ad.jp	133.4.3.6	Japan
archie.hana.nm.kr*	128.134.1.1	Korea
archie.sogang.ac.kr*	163.239.1.11	Korea
archie.uninett.no*	128.39.2.20	Norway
archie.rediris.es*	130.206.1.2	Spain
archie.luth.se*	130.240.18.4	Sweden
archie.switch.ch*	130.59.1.40	Switzerland
archie.ncu.edu.tw*	140.115.19.24	Taiwan
archie.doc.ic.ac.uk*	146.169.11.3	United Kingdom
archie.unl.edu	129.93.1.14	USA (NE)
archie.internic.net*	198.48.45.10	USA (NJ)
archie.rutgers.edu*	128.6.18.15	USA (NJ)
archie.ans.net	147.225.1.10	USA (NY)
archie.sura.net*	128.167.254.179	USA (MD)

Note: Sites marked with an asterisk "*" run archie version 3.0.

You can obtain details on using the electronic mail interface by sending mail to "archie" at any of the above server hosts. Put the word "help" as the text of your message for directions.

Questions, comments, and suggestions can be sent to the archie development group by sending mail to info@bunyip.com.

8.12 What is "gopher"?

The Internet Gopher presents an extremely wide variety of diverse types of information in an easy to use menu-driven interface. Gopher servers link information from all around the Internet in a manner that can be transparent to the user. (Users can easily discover the source of any piece of information, however, if they wish.) For example, gopher links databases of every type, applications, white pages directories, sounds, and pictures.

Some gophers are available via TELNET. Since most gophers are linked to other gophers, if you can get to one, you can get to many. You can, for example, telnet to naic.nasa.gov and use their public gopher.

The best way to use the gopher service, as with all client/server type services, is by running your own gopher client. The Internet Gopher was developed at the University of Minnesota. More information is available for anonymous FTP on the host boombox.micro.umn.edu.

8.13 What is the World Wide Web? What is Mosaic?

The World Wide Web is a distributed, hypermedia-based Internet information browser. It presents users with a friendly point and click interface to a wide variety of types of information (text, graphics, sounds, movies, etc.) and Internet services. It is possible to use the Web to access FTP archives, databases, and even gopher servers.

The most familiar implementations of the World Wide Web are the Mosaic clients developed by the National Center for Supercomputing Applications (NCSA). Mosaic software is available online at ftp.ncsa.uiuc.edu.

8.14 How do I find out about other Internet resource discovery tools?

The field of Internet resource discovery tools is one of the most dynamic on the Internet today. There are several tools in addition to those discussed here that are useful for discovering or searching Internet resources. The EARN (European Academic and Research Network) Association has compiled an excellent document that introduces many of these services and provides information about how to find out more about them. To obtain the document, send a message to listserv@earncc.bitnet or listserve%earncc.bitnet@cunyvm.cuny.edu. As the text of your message, type "GET filename" where the filename is either "nettools ps" or "nettols memo". The former is in PostScript format. This document is also available for anonymous FTP on some hosts, including naic.nasa.gov, where it is available in the files/general_info directory as earn-resource-tool-guide.ps and earn-resource-tool- guide.txt.

8.15 What is “TELNET“?

The term “TELNET” refers to the remote login that's possible on the Internet because of the TELNET Protocol [9]. The use of this term as a verb, as in “telnet to a host” means to establish a connection across the Internet from one host to another. Usually, you must have an account on the remote host to be able to login to it once you've made a connection. However, some hosts, such as those offering white pages directories, provide public services that do not require a personal account.

If your host supports TELNET, your command to connect to a remote host would probably be “telnet <hostname>“ or “telnet <host IP address>“. For example, “ telnet rs.internic.net” or “ telnet 198.41.0.5”.

9. Mailing Lists and Sending Mail

9.1 What is a mailing list?

A mailing list is an email address that stands for a group of people rather than for an individual. Mailing lists are usually created to discuss specific topics. Anybody interested in that topic, may (usually) join that list. Some mailing lists have membership restrictions, others have message content restrictions, and still others are moderated. Most “public” mailing lists have a second email address to handle administrative matters, such as requests to be added to or deleted from the list. All subscription requests should be sent to the administrative address rather than to the list itself!

9.2 How do I contact the administrator of a mailing list rather than posting to the entire list?

Today there are two main methods used by mailing list administrators to handle requests to subscribe or unsubscribe from their lists. The administrative address for many lists has the same name as the list itself, but with “-request” appended to the list name. So, to join the ietf-announce@cnri.reston.va.us list, you would send a message to ietf-announce- request@cnri.reston.va.us. Most often, requests to a “-request” mailbox are handled by a human and you can phrase your request as a normal message.

More often today, especially for lists with many readers, administrators prefer to have a program handle routine list administration. Many lists are accessible via LISTSERVE programs or other mailing list manager programs. If this is the case, the administrative address will usually be something like “listserv@host.domain”, where the address for the mailing list itself will be “list@host.domain”. The same listserve address can handle requests for all mailing lists at that host. When talking with a program, your subscription request will often be in the form, “subscribe ListName YourFirstName YourLastName” where you substitute the name of the list for ListName and add your real name at the end.

The important thing to remember is that all administrative messages regarding using, joining, or quitting a list should be sent to the administrative mailbox instead of to the whole list so that the readers of the list don't have to read them.

9.3 How do I send mail to other networks?

Mail to the Internet is addressed in the form user@host.domain. Remember that a domain name can have several components and the name of each host is a node on the domain tree. So, an example of an Internet mail address is june@nisc.sri.com.

There are several networks accessible via email from the Internet, but many of these networks do not use the same addressing conventions the Internet does. Often you must route mail to these networks through specific gateways as well, thus further complicating the address.

Here are a few conventions you can use for sending mail from the Internet to three networks with which Internet users often correspond.

Internet user to Internet user:
username@hostname.subdomain.toplevel domain e.g. gsmith@nisc.sri.COM

Internet user to BITNET user:
user%site.BITNET@BITNET-GATEWAY e.g.
gsmith%emoryu1.BITNET@cunyvm.cuny.edu.
gsmith%emoryu1@CORNELLC.CIT.CORNELL.EDU

Internet user to UUCP user:

user%host.UUCP@uunet.uu.net user%domain@uunet.uu.net

Internet user to SprintMail user:

/G=Mary/ S=Anderson/ O=co.abc/ ADMD=SprintMail/C=US/@SPRINT.COM
-or-
/PN=Mary.Anderson/ O=co.abc/ ADMD=SprintMail/C=US/@SPRINT.COM
(Case is significant.)

Internet user to CompuServe user:

Replace the comma in the CompuServe userid (represented here with x's) with a period, and add the compuserve.com domain name.

xxxx.xxxx@compuserve.com

CompuServe user to Internet user:

>Internet:user@host Insert > internet: before an Internet address.

Internet user to MCIMail user:

accountname@mcimail.com mci_id@mcimail.com
full_user_name@mcimail.com.

10. Miscellaneous “Internet lore” questions

10.1 What does :-) mean?

In many electronic mail messages, it is sometimes useful to indicate that part of a message is meant in jest. It is also sometimes useful to communicate emotion which simple words do not readily convey. To provide these nuances, a

collection of “smiley faces” has evolved. If you turn your head sideways to the left, :-) appears as a smiling face. Some of the more common faces are:

:-)	smile	:-(	frown
:)	also a smile	;-)	wink
:-D	laughing	8-)	wide-eyed
:-}	grin	:-X	close mouthed
:-]	smirk	:-o	oh, no!

10.2 What do “btw”, “fyi”, “imho”, “wrt”, and “rtfm” mean?

Often common expressions are abbreviated in informal network postings. These abbreviations stand for “by the way”, “for your information”, “in my humble [or honest] opinion”, “with respect to”, and “read the f*ing manual” (with the “f” word varying according to the vehemence of the reader :-).

10.3 What is the “FAQ” list?

This list provides answers to “Frequently Asked Questions” that often appear on various USENET newsgroups. The list is posted every four to six weeks to the news.announce.newusers group. It is intended to provide a background for new users learning how to use the news. As the FAQ list provide new users with the answers to such questions, it helps keep the newsgroups themselves comparatively free of repetition. Often specific newsgroups will have and frequently post versions of a FAQ list that are specific to their topics. The term FAQ has become generalized so that any topic may have its FAQ even if it is not a newsgroup.

Here is information about obtaining the USENET FAQs, courtesy of Gene Spafford:

Many questions can be answered by consulting the most recent postings in the news.announce.newusers and news.lists groups. If those postings have expired from your site, or you do not get news, you can get archived postings from the FTP server on the host rtfm.mit.edu.

These archived postings include all the Frequently Asked Questions posted to the news.answers newsgroups, as well as the most recent lists of Usenet newsgroups, Usenet-accessible mailing lists, group moderators, and other Usenet-related information posted to the news.announce.newusers and news.lists groups.

To get the material by FTP, log in using anonymous FTP (userid of anonymous and your email address as password).

The archived files, and FAQ files from other newsgroups, are all in the directory:

/pub/usenet/news.answers

Archived files from news.announce.newusers and news.lists are in:

/pub/usenet/news.announce.newusers
/pub/usenet/news.lists

respectively.

To get the information by mail, send an email message to: mail- server@pit-manager.mit.edu containing:

send usenet/ news.answers/TITLE/PART

where TITLE is the archive title, and PART is the portion of the posting you want.

Send a message containing "help" to get general information about the mail server, including information on how to get a list of archive titles to use in further send commands.

11. Suggested Reading

For further information about the Internet and its protocols in general, you may choose to obtain copies of the following works as well as some of the works listed as References:

Krol, Ed. (1992) *The Whole Internet User's Guide and Catalog*, 400 p. O'Reilly and Assoc., Inc. Sebastopol, CA.

Dern, Daniel P. (1993) *The Internet Guide for New Users*, 570 p. McGraw-Hill, Inc. New York, NY.

Fisher, Sharon. (1993) *Riding the Internet Highway*, 266 p. New Riders Publishing, Carmel, IN.

Frey, Donnalyn and Rick Adams. (1993) *!%@:: A Directory of Electronic Mail Addressing and Networks*, (third edition) 443 p. O'Reilly & Assoc., Inc. Sebastopol, CA.

Hoffman, Ellen and Lenore Jackson. (1993) *"FYI on Introducing the Internet: A Short Bibliography of Introductory Internetworking Readings for the Network Novice*", 4 p. (FYI 19/RFC 1463).

Kehoe, Brendan P. (1993) *Zen and the Art of the Internet: A Beginner's Guide*, (second edition) 112 p. Prentice Hall, Englewood Cliffs, NJ.

LaQuey, Tracy with Jeanne C. Ryer. (1992) *The Internet Companion: A Beginner's Guide to Global Networking*, 208 p. Addison-Wesley, Reading, MA.

Malkin, Gary, S. and Tracy LaQuey Parker. (1993) *"Internet Users' Glossary"*, 53 p. (FYI 18/RFC 1392).

Marine, April, et al. (1993) *Internet: Getting Started*, 360 p. Prentice Hall, Englewood Cliffs, NJ.

Martin, Jerry. (1993) *"There's Gold in them thar Networks! or Searching for Treasure in all the Wrong Places"*, 39 p. (FYI 10/RFC 1402).

Quarterman, John.(1993) *"Recent Internet Books"*, 15 p. (RFC 1432).

12. References

[1] **Reynolds, J., and J. Postel**, *"Assigned Numbers"*, STD 2, RFC 1340, USC/Information Sciences Institute, July 1992.

[2] **Postel, J.**, Editor, *"Internet Official Protocol Standards"*, STD 1, RFC 1540, Internet Architecture Board, October 1993.

[3] **Postel, J., and J. Reynolds**, *File Transfer Protocol (FTP)*, STD 9, RFC 959, USC/Information Sciences Institute, October 1985.

[4] **Postel, J**., *"Internet Protocol - DARPA Internet Program Protocol Specification"*, STD 5, RFC 791, DARPA, September 1981.

[5] **Postel, J.**, *"Transmission Control Protocol - DARPA Internet Program Protocol Specification"*, STD 7, RFC 793, DARPA, September 1981.

[6] **Leiner, B., Cole, R., Postel, J., and D. Mills**, *"The DARPA Internet Protocol Suite"*, IEEE INFOCOM85, Washington D.C., March 1985. Also in IEEE Communications Magazine, March 1985. Also as ISI/RS-85-153.

[7] **Cerf, V.**, *"The Internet Activities Board"*, RFC 1160, CNRI, May 1990.

[8] **Postel, J.**, *"Simple Mail Transfer Protocol"*, STD 10, RFC 821, USC/Information Sciences Institute, August 1982.

[9] **Postel, J., and J. Reynolds**, *"TELNET Protocol Specification"*, STD 8, RFC 854, USC/Information Sciences Institute, May 1983.

[10] **Postel, J.**, *"Instructions to RFC Authors"*, RFC 1543, USC/Information Sciences Institute, October 1993.

[11] **Malkin, G., Marine, A., and J. Reynolds**, *"FYI on Questions and Answers: Answers to Commonly Asked 'Experienced Internet User' Questions"*, FYI 7, RFC 1207, FTP Software, SRI, USC/Information Sciences Institute, February 1991.

[12] **Postel, J.**, *"Introduction to the STD Notes"*, RFC 1311, USC/Information Sciences Institute, March 1992.

[13] **Krol, E., and E. Hoffman**, *"FYI on 'What is the Internet?'"*, FYI 20, RFC 1462, University of Illinois, Merit Network, Inc., May 1993.

13. Condensed Glossary

As with any profession, computers have a particular terminology all their own. Below is a condensed glossary to assist in making some sense of the Internet world.

ACM Association for Computing Machinery. A group established in 1947 to promote professional development and research on computers.

address There are three types of addresses in common use within the Internet. They are email address; IP, internet or Internet address; and hardware or MACINTOSH address. An electronic mail address is the string of characters that you must give an electronic mail program to direct a message to a particular person. A MACINTOSH address is the hardware address of a device connected to a shared media. See " internet address" for its definition.

AI Artificial Intelligence. The branch of computer science that deals with the simulation of human intelligence by computer systems.

AIX Advanced Interactive Executive — IBM's version of Unix.

ANSI American National Standards Institute. This organization is responsible for approving U.S. standards in many areas, including computers and communications. Standards approved by this organization are often called ANSI standards (e.g., ANSI C is the version of the C language approved by ANSI). ANSI is a member of ISO. See also: International Organization for Standardization.

ARP Address Resolution Protocol. Used to dynamically discover the low level physical network hardware address that corresponds to the high level IP address for a given host. ARP is limited to physical network systems that support broadcast packets that can be heard by all hosts on the network. It is defined in STD 37, RFC 826.

ARPA Advanced Research Projects Agency. An agency of the U.S. Department of Defense responsible for the development of new technology for use by the military. ARPA was responsible for funding much of the development of the Internet we know today, including the Berkeley version of Unix and TCP/IP.

ARPANET Advanced Research Projects Agency Network. A pioneering longhaul network funded by ARPA. It served as the basis for early networking research as well as a central backbone during the development of the Internet. The ARPANET consisted of individual packet switching computers interconnected by leased lines.

AS Autonomous System. A collection of routers under a single administrative authority using a common Interior Gateway Protocol for routing packets.

ASCII American (National) Standard Code for Information Interchange. A standard character-to-number encoding widely used in the computer industry.

B Byte. One character of information, usually eight bits wide.

b bit - binary digit. The smallest amount of information which may be stored in a computer.

BBN Bolt Beranek and Newman, Inc. The Cambridge, MA company responsible for development, operation and monitoring of the ARPANET, and later, the Internet core gateway system, the CSNET Coordination and Information Center (CIC), and NSFNET Network Service Center (NNSC).

BITNET An academic computer network that provides interactive electronic mail and file transfer services, using a store-and-forward protocol, based on IBM Network Job Entry protocols. BITNET-II encapsulates the BITNET protocol within IP packets and depends on the Internet to route them. There are three main constituents of the network: BITNET in the United States and Mexico, NETNORTH in Canada, and EARN in Europe. There are also AsiaNet, in Japan, and connections in South America. See CREN.

bps bits per second. A measure of data transmission speed.

BSD Berkeley Software Distribution. Implementation of the UNIX operating system and its utilities developed and distributed by the University of California at Berkeley. "BSD" is usually preceded by the version number of the distribution, e.g., "4.3 BSD" is version 4.3 of the Berkeley UNIX distribution. Many Internet hosts run BSD software, and it is the ancestor of many commercial UNIX implementations.

catenet A network in which hosts are connected to networks with varying characteristics, and the networks are interconnected by gateways (routers). The Internet is an example of a catenet.

CCITT International Telegraph and Telephone Consultative Committee. This organization is part of the United National International Telecommunications Union (ITU) and is responsible for making technical recommendations about telephone and data communications systems.

core gateway Historically, one of a set of gateways (routers) operated by the Internet Network Operations Center at BBN. The core gateway system forms a central part of Internet routing in that all groups had to advertise paths to their networks from a core gateway.

CREN The Corporation for Research and Educational Networking. This organization was formed in October 1989, when BITNET and CSNET (Computer + Science NETwork) were combined under one administrative authority. CSNET is no longer operational, but CREN still runs BITNET. See also: BITNET.

DARPA See ARPA.

Datagram A self-contained, independent entity of data carrying sufficient information to be routed from the source to the destination computer without reliance on earlier exchanges between this source and destination computer and the transporting network.

DCA Defense Communications Agency. Former name of the Defense Information Systems Agency (DISA). See DISA.

DDN Defense Data Network. A global communications network serving the US Department of Defense composed of MILNET, other portions of the Internet, and classified networks which are not part of the Internet. The DDN is used to connect military installations and is managed by the Defense Information Systems Agency (DISA). See also: DISA.

DDN NIC The Defense Data Network Network Information Center. The network information center at Network Solutions, Inc., funded by DISA, that provides information services to the DDN community. It is also a primary repository for RFCs, and a delegated registration authority for military networks.

DEC Digital Equipment Corporation

DECnet Digital Equipment Corporation network. A proprietary network protocol designed by Digital Equipment Corporation. The functionality of each Phase of the implementation, such as Phase IV and Phase V, is different.

default route A routing table entry which is used to direct packets addressed to networks not explicitly listed in the routing table.

DISA Defense Information Systems Agency Formerly. called DCA, this is the government agency responsible for installing the Defense Data Network (DDN) portion of the Internet, including the MILNET lines and nodes. Currently, DISA administers the DDN, and supports the user assistance services of the DDN NIC.

DNS The Domain Name System is a general purpose distributed, replicated, data query service. The principal use is the lookup of host IP addresses based on host names. The style of host names now used in the Internet is called "domain name", because they are the style of names used to look up anything in the DNS. Some important domains are: .COM (commercial), .EDU (educational), .NET (network operations), .GOV (U.S. government), and .MIL (U.S. military). Most countries also have a domain. For example, .US (United States), .UK (United Kingdom), .AU (Australia). It is defined in STD 13, RFCs 1034 and 1035.

DOD U.S. Department of Defense

DOE U.S. Department of Energy

dot address (dotted address notation) Dot address. refers to the common notation for IP addresses of the form A.B.C.D; where each letter represents, in decimal, one byte of a four byte IP address.

Dynamic Adaptive Routing Automatic rerouting of traffic based on a sensing and analysis of current actual network conditions. NOTE: this does not include cases of routing decisions taken on predefined information.

EARN European Academic Research Network.

EBCDIC Extended Binary-coded Decimal Interchange Code. A standard character-to-number encoding used primarily by IBM computer systems. See also: ASCII.

EGP Exterior Gateway Protocol. A protocol which distributes routing information to the routers which connect autonomous systems. The term "gateway" is historical, as "router" is currently the preferred term. There is also a routing protocol called EGP defined in STD 18, RFC 904.

Ethernet A 10-Mb/s standard for LANs, initially developed by Xerox, and later refined by Digital, Intel and Xerox (DIX). All hosts are connected to a coaxial cable where they contend for network access using a Carrier Sense Multiple Access with Collision Detection (CSMA/CD) paradigm.

FDDI Fiber Distributed Data Interface. A high-speed (100Mb/s) LAN standard. The underlying medium is fiber optics, and the topology is a dual-attached, counter-rotating token ring.

FIPS Federal Information Processing Standard.

FTP File Transfer Protocol. A protocol which allows a user on one host to access, and transfer files to and from, another host over a network. Also, FTP is usually the name of the program the user invokes to execute the protocol. It is defined in STD 9, RFC 959.

gateway See router.

GB Gigabyte. A unit of data storage size which represents 10^9 (one billion) characters of information.

Gb Gigabit. 10^9 bits of information (usually used to express a data transfer rate; as in, 1 gigabit/second = 1Gbps).

GNU Gnu's Not UNIX. A UNIX-compatible operating system developed by the Free Software Foundation.

header The portion of a packet, preceding the actual data, containing source and destination addresses, and error checking and other fields. A header is also the part of an electronic mail message that precedes the body of a message and contains, among other things, the message originator, date and time.

HP Hewlett-Packard.

I/O Input/Output.

IAB Internet Architecture Board. The technical body that oversees the development of the Internet suite of protocols. It has two task forces: the IETF and the IRTF.

IBM International Business Machines Corporation.

ICMP Internet Control Message Protocol. ICMP is an extension to the Internet Protocol. It allows for the generation of error messages, test packets and informational messages related to IP. It is defined in STD 5, RFC 792.

IEEE Institute for Electrical and Electronics Engineers.

IETF Internet Engineering Task Force. The IETF is a large open community of network designers, operators, vendors, and researchers whose purpose is to coordinate the operation, management and evolution of the Internet, and to resolve short- and mid-range protocol and architectural issues. It is a major source of proposed protocol standards which are submitted to the Internet Engineering Steering Group for final approval. The IETF meets three times a year and extensive minutes of the plenary proceedings are issued.

internet internetwork While an internet is a network, the term "internet" is usually used to refer to a collection of networks interconnected with routers.

Internet The Internet (note the capital "I") is the largest internet in the world. Is a three level hierarchy composed of backbone networks (e.g., NSFNET, MILNET), mid-level networks, and stub networks. The Internet is a multiprotocol internet.

internet address The 32-bit address defined by the Internet Protocol in STD 5, RFC 791. It is usually represented in dotted decimal notation. An internet, or IP, address uniquely identifies a node on an internet.

IP Internet Protocol. The Internet Protocol, defined in STD 5, RFC 791, is the network layer for the TCP/IP Protocol Suite. It is a connectionless, best-effort packet switching protocol.

IRTF Internet Research Task Force. The IRTF is chartered by the IAB to consider long-term Internet issues from a theoretical point of view. It has Research Groups, similar to IETF Working Groups, which are each tasked to

discuss different research topics. Multi-cast audio/video conferencing and privacy enhanced mail are samples of IRTF output.

ISO International Organization for Standardization. A voluntary, nontreaty organization founded in 1946 which is responsible for creating international standards in many areas, including computers and communications. Its members are the national standards organizations of the 89 member countries, including ANSI for the U.S.

KB Kilobyte. A unit of data storage size which represents 10^3 (one thousand) characters of information.

Kb Kilobit. 10^3 bits of information (usually used to express a data transfer rate; as in, 1 kilobit/second = 1Kbps = 1Kb).

LAN Local Area Network. A data network intended to serve an area of only a few square kilometers or less. Because the network is known to cover only a small area, optimizations can be made in the network signal protocols that permit data rates up to 100Mb/s.

LISP List Processing Language. A high-level computer language invented by Professor John McCarthy in 1961 to support research into computer based logic, logical reasoning, and artificial intelligence. It was the first symbolic (as opposed to numeric) computer processing language.

MAC Medium Access Control. The lower portion of the datalink layer. The MAC differs for various physical media.

Macintosh Apple Macintosh computer.

MAN Metropolitan Area Network. A data network intended to serve an area approximating that of a large city. Such networks are being implemented by innovative techniques, such as running fiber cables through subway tunnels. A popular example of a MAN is SMDS.

MB Megabyte. A unit of data storage size which represents 10^6 (one million) characters of information.

Mb Megabit. 10^6 bits of information (usually used to express a data transfer rate; as in, 1 megabit/second = 1Mbps).

MILNET Military Network. A network used for unclassified military production applications. It is part of the DDN and the Internet.

MIT Massachusetts Institute of Technology.

MTTF Mean Time to Failure. The average time between hardware breakdown or loss of service. This may be an empirical measurement or a calculation based on the MTTF of component parts.

MTTR Mean Time to Recovery (or Repair). The average time it takes to restore service after a breakdown or loss. This is usually an empirical measurement.

MVS Multiple Virtual Storage. An IBM operating system based on OS/1.

NASA National Aeronautics and Space Administration.

NBS National Bureau of Standards Now called NIST.

network number The network portion of an IP address. For a class A network, the network address is the first byte of the IP address. For a class B network, the network address is the first two bytes of the IP address. For a class C network, the network address is the first three bytes of the IP address. In each case, the remainder is the host address. In the Internet, assigned network addresses are globally unique.

NFS Network File System. A protocol developed by Sun Microsystems, and defined in RFC 1094, which allows a computer system to access files over a network as if they were on its local disks. This protocol has been incorporated in products by more than two hundred companies, and is now a de facto Internet standard.

NIC Network Information Center. A organization that provides information, assistance and services to network users.

NOC Network Operations Center. A location from which the operation of a network or internet is monitored. Additionally, this center usually serves as a clearinghouse for connectivity problems and efforts to resolve those problems.

NIST National Institute of Standards and Technology. United States governmental body that provides assistance in developing standards. Formerly the National Bureau of Standards (NBS).

NSF National Science Foundation. A U.S. government agency whose purpose is to promote the advancement of science. NSF funds science researchers, scientific projects, and infrastructure to improve the quality of scientific research. The NSFNET, funded by NSF, is an essential part of academic and research communications.

NSFNET National Science Foundation Network. The NSFNET is a high speed "network of networks" which is hierarchical in nature. At the highest level is a backbone network which spans the continental United States. Attached to that are mid-level networks and attached to the mid-levels are campus and local networks. NSFNET also has connections out of the U.S. to Canada, Mexico, Europe, and the Pacific Rim. The NSFNET is part of the Internet.

NSFNET Mid-level Level Network A network connected to the highest level of the NSFNET that covers a region of the United States. It is to mid-level networks that local sites connect. The mid-level networks were once called "regionals".

OSI Open Systems Interconnection. A suite of protocols, designed by ISO committees, to be the international standard computer network architecture.

OSI Reference Model A seven-layer structure designed to describe computer network architectures and the way that data passes through them. This model was developed by the ISO in 1978 to clearly define the interfaces in multivendor networks, and to provide users of those networks with conceptual guidelines in the construction of such networks.

OSPF Open Shortest-Path First. Interior Gateway Protocol A link state, as opposed to distance vector, routing protocol. It is an Internet standard IGP defined in RFC 1247.

packet The unit of data sent across a network. "Packet" a generic term used to describe unit of data at all levels of the protocol stack, but it is most correctly used to describe application data units.

PC Personal Computer.

PCNFS Personal Computer Network File System.

PPP Point-to-Point Protocol. The Point-to-Point Protocol, defined in RFC 1548, provides a method for transmitting packets over serial point-to-point links.

protocol A formal description of message formats and the rules two computers must follow to exchange those messages. Protocols can describe low-level details of machine-to-machine interfaces (e.g., the order in which bits and bytes are sent across a wire) or high-level exchanges between allocation programs (e.g., the way in which two programs transfer a file across the Internet).

RFC The document series, begun in 1969, which describes the Internet suite of protocols and related experiments. Not all (in fact very few) RFCs describe Internet standards, but all Internet standards are written up as RFCs.

RIP Routing Information Protocol. A distance vector, as opposed to link state, routing protocol. It is an Internet standard IGP defined in STD 34, RFC 1058 (updated by RFC 1388).

RJE Remote Job Entry. The general protocol for submitting batch jobs and retrieving the results.

router A device which forwards traffic between networks. The forwarding decision is based on network layer information and routing tables, often constructed by routing protocols.

RPC Remote Procedure Call. An easy and popular paradigm for implementing the client-server model of distributed computing. In general, a request is sent to a remote system to execute a designated procedure, using arguments supplied, and the result returned to the caller. There are many variations and subtleties in various implementations, resulting in a variety of different (incompatible) RPC protocols.

server A provider of resources (e.g., file servers and name servers).

SLIP Serial Line Internet Protocol. A protocol used to run IP over serial lines, such as telephone circuits or RS-232 cables, interconnecting two systems. SLIP is defined in STD 47, RFC 1055.

SMTP Simple Mail Transfer Protocol. A protocol, defined in STD 10, RFC 821, used to transfer electronic mail between computers. It is a server to server protocol, so other protocols are used to access the messages.

SNA Systems Network Architecture. A proprietary networking architecture used by IBM and IBM-compatible mainframe computers.

SNMP Simple Network Management Protocol. The Internet standard protocol, defined in STD 15, RFC 1157, developed to manage nodes on an IP network. It is currently possible to manage wiring hubs, toasters, jukeboxes, etc.

subnet A portion of a network, which may be a physically independent network, which shares a network address with other portions of the network and is distinguished by a subnet number. A subnet is to a network what a network is to an internet.

subnet number A part of the internet address which designates a subnet. It is ignored for the purposes internet routing, but is used for intranet routing.

T1 An AT&T term for a digital carrier facility used to transmit a DS-1 formatted digital signal at 1.544 megabits per second.

T3 A term for a digital carrier facility used to transmit a DS-3 formatted digital signal at 44.746 megabits per second.

TCP Transmission Control Protocol. An Internet Standard transport layer protocol defined in STD 7, RFC 793. It is connection-oriented and stream-oriented, as opposed to UDP.

TCP/IP Transmission Control Protocol/Internet Protocol. This is a common shorthand which refers to the suite of application and transport protocols which run over IP. These include FTP, TELNET, SMTP, and UDP (a transport layer protocol).

Telenet A public packet switched network using the CCITT X.25 protocols. It should not be confused with Telnet.

TELNET Telnet is the Internet standard protocol for remote terminal connection service. It is defined in STD 8, RFC 854 and extended with options by many other RFCs.

Token Ring A token ring is a type of LAN with nodes wired into a ring. Each node constantly passes a control message (token) on to the next; whichever node has the token can send a message. Often, “Token Ring” is used to refer to the IEEE 802.5 token ring standard, which is the most common type of token ring.

Tymnet A public character-switching/packet-switching network operated by British Telecom.

UDP User Datagram Protocol. An Internet Standard transport layer protocol defined in STD 6, RFC 768. It is a connectionless protocol which adds a level of multiplexing to IP.

ULTRIX UNIX-based operating system for Digital Equipment Corporation computers.

UNIX An operating system developed by Bell Laboratories that supports multiuser and multitasking operations.

UUCP UNIX-to-UNIX Copy Program. This was initially a program run under the UNIX operating system that allowed one UNIX system to send files to another UNIX system via dial-up phone lines. Today, the term is more commonly used to describe the large international network which uses the UUCP protocol to pass news and electronic mail.

VMS Virtual Memory System. A Digital Equipment Corporation operating system.

WAN Wide Area Network. A network, usually constructed with serial lines, which covers a large geographic area.

WHOIS An Internet program which allows users to query databases of people and other Internet entities, such as domains, networks, and hosts. The information for people generally shows a person's company name, address, phone number and email address.

XNS Xerox Network System. A network developed by Xerox corporation. Implementations exist for both 4.3BSD derived systems, as well as the Xerox Star computers.

X.25 A data communications interface specification developed to describe how data passes into and out of public data communications networks. The CCITT and ISO approved protocol suite defines protocol layers 1 through 3.

14. Security Considerations

Security issues are not discussed in this memo.

15. Authors' Addresses

April N. Marine
Network Applications and Information Center
NASA Ames Research Center M/S 204-14
Moffett Field, CA 94035-1000
Phone: (415) 604-0762
EMail: amarine@atlas.arc.nasa.gov

Joyce K. Reynolds
USC/Information Sciences Institute
4676 Admiralty Way, Suite 1001
Marina del Rey, CA 90292-6695
Phone: (310) 822-1511
EMail: jkrey@isi.edu

Gary Scott Malkin
Xylogics, Inc.
53 Third Avenue
Burlington, MA 01803
Phone: (617) 272-8140
EMail: gmalkin@Xylogics.COM

INDEX

E

F

G

H

I

K

L

M

N

O

P

Q

R

S

T

U

V

W

X